APRILE '85
ANNO V - N°
39
microcomputer
46
28
60
54
72

I KIT DI microcomputer

APPLE-minus
per aggiungere le minuscole ai vostro Apple II

M/1: Eprom programmata per Apple II delle nuove serie (rev 7 e successive) **L. 30.000**

M/2: Eprom programmata per Apple II delle serie precedenti la 7 + circuito stampato + 2 zoccoli 24 pin + 1 zoccolo 16 pin **L. 40.000**

M/3: come il kit M/2, basetta montata e collaudata **L. 55.000**

Descrizione: MC n. 3 - 4 - 5 - 7

TAVOLETTA GRAFICA
per Apple II

Si collega allo zoccolo dei paddle dell'Apple II e consente di disegnare sullo schermo in alta risoluzione. È fornita montata, calibrata e collaudata; è compreso il piano di lavoro con il menu su foglio di cartoncino plastificato e un minifloppy con tutto il software, sia in Applesoft sia compilato **L. 215.000**

Descrizione: MC n. 8 - 9 - 10 - 11 - 13

VIC-TRISLOT
per Commodore VIC-20

Si collega allo slot del VIC-20 e consente di installare tre cartucce. È costituito da un circuito stampato doppia faccia su vetronite, con fori metallizzati e pettine dorato, tre connettori (già saldati) professionali con contatti dorati per l'inserimento delle schede, piedini sul fondo della basetta. **L. 60.000**

Descrizione: MC n. 16

Per acquistare i nostri kit:
Il pagamento può essere effettuato tramite conto corrente postale n. 14414007 intestato a Technimedia s.r.l. via Valsolda 135 00141 Roma o vaglia postale. Per una maggiore rapidità puoi inviarci una lettera con allegato assegno di c/c bancario o circolare intestato a Technimedia s.r.l. Infine puoi acquistarlo direttamente presso i nostri uffici di Roma o al nostro stand in occasione delle mostre.
N.B. Specificare nell'ordine (indicando il numero di partita IVA) se desideri ricevere la fattura.

INDICE DEGLI INSERZIONISTI

Anno V - numero 39
marzo 1985
L. 4.000
Direttore:
Paolo Nuti
Condirettore:
Marco Marinacci
Ricerca e sviluppo:
Bo Arnklit
Collaboratori:
Maurizio Bergami, Raffaello De Masi, Andrea de Prisco, Valter Di Dio, Corrado Giustozzi, Fabio Marzocca, Alberto Morando, Tommaso Pantuso, Pierluigi Panunzi, Francesco Petroni, Gina Principi, Fabio Schiattarella, Leo Sorge, Pietro Tasso
Segreteria di redazione:
Paola Pujia (responsabile), Giovanna Molinari
Roberto Rubino
Grafica e impaginazione:
Roberto Saltarelli
Grafica copertina:
Studio Azeta - Roma
Fotografia:
Dario Tassa
Amministrazione:
Maurizio Ramaglia (responsabile), Anna Rita Fratini, Pina Salvatore
Abbonamenti ed arretrati:
Matteo Piemontese
Direttore Responsabile:
Marco Marinacci

MCmicrocomputer è una pubblicazione Technimedia
Via Valsolda 135 00141 Roma
Tel 06/898 654-899 526
Registrazione del Tribunale di Roma n. 298/81 dell'11 agosto 1981

Pubblicità:
Technimedia,
Via Valsolda 135,
00141 Roma,
tel 06/898 654-899 526
Produzione pubblicitaria
Cesare Veneziani
Abbonamento a 12 numeri.
Italia L. 40 000 Europa e paesi del bacino mediterraneo (spedizione via aerea) L. 84 000
Americhe, Giappone, Asia etc.
L. 120 000 (spedizione via aerea)
C/c postale n. 14414007 intestato a Technimedia s.r.l. Via Valsolda, 135 00141 Roma
Composizione e fotolito:
Start Photolito Via Acuto 137, GRA km 29 Roma
Stampa:
Grafiche P.F.G. Via Trasporti 46/48 00040 Ariccia (Roma)
Zona Industriale Nettunense
Concessionaria per la distribuzione:
Parrini & C. Roma P.zza Indipendenza 11b - Cent. Tel. 4940841

1985 - Anno V
marzo, n. 3, mensile

Associato USPI

Scuola: ma dategli il word processor

Scavalcati da un'orda di ragazzini assatanati, istituzioni, insegnanti ed esperti di pedagogia, arrancano nel tentativo di correre ai ripari.

Se solo tre anni orsono, quello del computer nella scuola elementare poteva essere un argomento di punta, oggi è praticamente banale basta aprire il giornale per trovare, ogni giorno, notizia di un qualche convegno, mostra, incontro o conferenza sul tema "Il computer nella scuola".

Come direttore di MCmicrocomputer dovrei rallegrarmi di tanto interesse. E me ne rallegro, infatti, ma solo in parte. E per due motivi prima di tutto, dal momento che la previsione era talmente facile e banale da essere scritta sulle riviste specializzate di mezzo monda (compresa la nostra), sarebbe stato bene che la scuola si fosse occupata estesamente del fenomeno con un certo anticipo, e non ad informatizzazione infantile di massa ormai avvenuta. A porte l'inutilità di piangere sul latte versato, non è però il ritardo con cui la scuola sta affrontando il fenomeno, piuttosto la mentalità con cui lo sta affrontando.

Signori, la necessità di provvedere con urgenza alla alfabetizzazione informatica di massa è ormai accettata, ed anzi, scontata. Con la complicità dei genitori che, superata l'iniziale titubanza, hanno comprato sempre più freneticamente il computer ai loro figlioli, ci troviamo di fronte ad una generazione di piccoli maestri della tastiera, che con il computer si divertono, che sempre più spesso utilizzano il computer con profitto, che del computer tutto hanno, tranne che paura. Bene, non benissimo, perche molti dei piccoli utenti di Commodore, Spectrum etc., etc., non hanno mai aperto un manuale: l'informaziane sul come fare questo o quello si trasmette tra i banchi per via orale. Vero è che Iliade ed Odissea si sono tramandate per via orale, ma altrettanto vero è che la cultura ed il progresso si trasmettono assai meglio per scritto. Insomma, ci sarebbe bisogno di studiare computer, un po' più sistematicamente, sin dai primi anni di scuola.

Sapete invece qual è l'argomento principe dei convegni che si occupano di informatico nella scuola? L'impiego del computer come sussidio didattico. Ben venga il computer per concretizzare qualcosa prima irrealizzabile, per esempio la visualizzazione dinamica di una famiglia di curve o l'evolversi della volta celeste ad una data latitudine (senza bisogno di andare al non sempre raggiungibile o efficente planetario), ben venga il computer per la simulazione numerica e/o grafica di fenomeni fisici, chimici, economici etc., etc. Ma sapete invece di cosa si occupa gran parte del software didattico di cui tanto si discute? Di somministrazione nozionistica e di test. Se un giochino geografico può ancora essere utile per mandare a mente nozioni, come nomi e localizzazioni geografiche che necessariamente non possono che essere imparate a memoria, perché dovrei imparare la matematica leggendo *le regole dei prodotti notevoli su una scomoda schermata anzichè sulla comoda pagina di un libro? E perche dovrei affaticarmi a leggere sullo schermo le equazioni da risolvere per esercizia anziche comodamente su un libro, quando il computer domestico non è e non può per il momento essere in grado di far girare programmi tanto sofisticati da individuare il singolo errore commesso nel procedimento di soluzione? Ma non è più semplice controllare il risultato dell'esercizio sul libro anziche immetterlo da tastiera?*

Poveri ragazzi, erano convinti che il computer fosse facile utile e divertente, e c'è chi vuole trasformarlo in uno strumento repressivo e noioso. Per fortuna, come osservo un collega di fronte ad una gara di videogiochi andata deserta a causa di un regolamento farraginoso e repressivo, "i ragazzini non sono mica scemi". E così si scopre che a 8 anni hanno imparato ad usare il word processor di papà. E divertendosi ad usare il word processor, superata la noia di non dover pasticciare in brutta e poi ricopiare, hanno preso gusto a scrivere. E alla fine tanta è diventata l'abitudine a scrivere frasi sintatticamente corrette, che (senza word processor) viene bene anche il tema in classe.

Insomma questo computer nella scuola vogliamo utilizzarlo per le sue prerogative di potente strumento atto a sviluppare le capacità di chi lo usa, o come potente strumento per evitare che i bambini appaiano più bravi degli insegnanti?

Paolo Nuti

Due minifloppy
Dysan per te
se ti abboni a
microcomputer
Dysan
Convenience Pack
2 Mini Diskettes
100% Error Free
Dysan
Dysan
5 1/4 Flexible
Diskettes
Discover
The Dysan
Difference
Se ti abboni o rinnovi l'abbonamento
a MCmicrocomputer, puoi ricevere una confezione
di due minifloppy Dysan, singola faccia doppia densità,
con un supplemento di sole 3.500 lire.
Non perdere quest'occasione!
Ritaglia e spedisci oggi stesso il tagliando
per sottoscrivere l'abbonamento pubblicato
nell'ultima pagina della rivista.
I minifloppy ti saranno spediti in una robusta
confezione a prova di danneggiamenti postali.
I prodotti Dysan sono distribuiti in Italia dalla
Datamatic, Via Volturno 46, 20124 Milano

Un programma rubato

Spett.le Redazione,

mi duole comunicare che sul numero 36 della Rivista MC, alle pagine 173/174 a stato pubblicato a nome di un certo signor Enrico Elia un progamma Basic ("carie") evidentemente plagiato Tale listato a praticamente identico, infatti, a quello pubblicato sul libro "Tecniche avanzate in Assembler" alle pagine 155/158 della edizione italiana di Mc-Graw Hill. Tale programma era già stato commercializzato circa due anni fa da una software house inglese sotto il nome "Cross", autore lo scrittore del libro S. Nicholls

Se mi a permesso dare un suggerimento ai redattori che selezionano i programmi da pubblicare tra quelli inviati dai lettori consiglio di diffidare della autenticita ed originalita dei programmi inviati da lettori italiani che contengono testi in inglese, come il programma "Carie" (lives, score, home...)

Visto che il signor Bergami, oltre a non sapere fare i test delle macchine nuove (vedi la pessima prova del QL a molte altre), si mostra piuttosto ingenuo anche in questo campo farebbe meglio a darsi all'ippica.

Distinti saluti

Luigi Callegari - Sumirago (VA)

Interpellato in merito, Maurizio Bergami mi ha risposto che a lui in realtà l'ippica piace molto e che, anzi, solo per mancanza di tempo ha smesso di andare a cavallo Al di la di questo e del fatto che ci vuole una certa maleducazione per esprimersi nei termini che lei usa, signor Callegari, la sua lettera solleva un problema degno di interesse

Andiamo per punti. grazie innanzi tutto per la segnalazione, che a dire il vero non ho controllato ma prendo per buona, tanto ormai il programma a stato pubblicato (e pagato al sedicente autore) e non a possibile ne cancellarlo da tutte le riviste né chiedere i soldi indietro al signor Elia. Purtroppo può succedere e, mi creda, me ne dispiace molto, perché simili episodi finiscono per danneggiare l'immagine di MCmicrocomputer e non mi sembra intelligente che ci siano lettori che, per pochi soldi e probabilmente la brama di avere il nome pubblicato sulla rivista, cerchino di carpire la buona fede del redattore di turno e a volte, come in questo caso, riuscendoci, nonostante il fatto che un po' tutti noi seguiamo abbastanza le varie pubblicazioni, estere in particolare, e un buon numero di.. furbi vengono fermati. Ovviamente, a impossibile che ogni redattore legga tutte le riviste,e peggio, ricordi tutto di tutte a per sempre... A proposito, un altro lettore ci ha fatto fessi. il programma "Dispersione termica" di Marco Mastri di Ancona per il VIC, pubblicato nel numero scorso, era gia apparso sull'enciclopedia Basic di Curcio

Quanto al suo consiglio per non essere fatti fessi, sappia che purtroppo il suo sistema non funziona, sono molte, direi la maggior parte, le persone che inseriscono termini inglesi nei loro programmi. E, d'altra parte, uno che ruba un programma da una rivista inglese e non vuole farsi beccare, la prima cosa che fa è spesso proprio quella di togliere le parole inglesi. Grazie, ripeto, ma il consiglio e un po' ingenuo per cercare di stanare per quanto possibile i furbi usiamo altri trucchi, che mi consentira di non rivelare sulla rivista. Devo comunque aggiungere che in linea di massima abbiamo una discreta fiducia nella massa dei nostri lettori, che non ci fa dubitare di proprio tutti i programmi appena passabili che arrivano ed in effetti i casi "spiacevoli" sono piuttosto isolati.

Non torniamo sul discorso della pessima prova del QL. i lettori piu attenti si ricorderanno di lei e della sua lettera pubblicata nel numero 33, nella cui risposta Bergami le ha spiegato di come fossero infondate certe sue affermazioni. Devo pero dire che un passo avanti c'e stato in quella occasione lei ci ha praticamente accusati di vendere fumo, non avendo neanche acceso il QL, questa volta ci da semplicemente degli ingenui, ma sembra almeno non mettere in dubbio la nostra buona fede. Chissa se, in un'ipotetica sua terza lettera, lei non riuscirà a limitarsi ad un discorso anche critico, ma civile ed educato E quindi, mi scusi, ben piu gradito

Marco Marinacci

Commodore Plus/4

4 programmi dentro

la pianificazione finanziaria; un DATABASE, per la creazione e gestione di archivi; un pacchetto di BUSINESS GRAPHICS, per visualizzare i tuoi dati sotto forma di diagrammi e istogrammi.

È facile: premi un tasto e puoi richiamare subito i programmi che vuoi, senza dover aspettare il tempo di caricamento. Puoi anche usarli insieme, perchè sono integrati (e lo schermo è divisibile in 4 parti).

Commodore Plus/4 è lo strumento ideale in campo finanziario e gestionale, per la scuola e per la professione.

Puoi aggiungere tanti altri programmi e lo usi con facilità per risolvere un'infinità di complessi problemi.

E hai chiesto il prezzo? Straordinario (se pensi che i 4 programmi incorporati valgono da soli ben di più).

Commodore Italiana S.p.A.
tel. 02/618321.

commodore COMPUTER

Per chi ha problemi con l'ADP Basic

Rispondiamo alle numerose lettere ricevute sull'ADP Basic, il tool di istruzioni per il Commodore 64 orientato alla gestione facilitata delle periferiche che si conclude su questo numero

Il primo articolo della serie è apparso sul numero 35 di MC, ben 5 mesi fa, il programma è piuttosto ponderoso e la sua digitazione può obiettivamente presentare delle difficoltà.

Precisiamo che il programma, così come è, funziona e che le linee che nel numero 35 erano risultate illegibili per un problema della tipografia sono state ripubblicate nel numero successivo

In ogni caso, nell'articolo pubblicato in questo stesso numero potete trovare ripubblicato l'intero ADP Basic al quale a questo punto, se avete dei problemi, consigliamo di riferirvi. Precisiamo che la diversità di questa "veste finale" consiste solo nel fatto che vi è un solo ciclo FOR. NEXT per il caricamento di tutti i DATA dei 4096 byte occupati. Una volta digitato tutto il programma e mandatolo in esecuzione con RUN, non è necessario altro tutti i nuovi comandi sono disponibili da tastiera alla stessa maniera dei comandi del Basic standard. Basta digitarli. se ad esempio vogliamo il catalog di un dischetto digiteremo CAT e Return allo stesso modo di come digiteremmo LIST e Return per conoscere il listato di un programma Basic.

Quindi, ricapitolando, se dopo aver battuto l'intero listato dell'ADP Basic pubblicato su questo numero o le varie porzioncine dei numeri scorsi, alla richiesta CAT (come qualcuno ci ha scritto) si blocca tutto, significa che avete compiuto qualche errore di copiatura, controllate bene (e sottolineiamo, bene) e vi accorgerete di aver dimenticato qualcosa.

Per finire, qualche precisazione per i lettori che hanno richiesto o hanno intenzione di richiedere il dischetto con l'ADP Basic (operazione che consigliamo vivamente a chi non riesce a digitare correttamente il listato). Sul dischetto sono presenti sia la versione per la stampante 802 che quella per la 801.

A proposito di quest'ultima, vogliamo ricordare che parlare di 801 è la stessa cosa che parlare di VIC-1515, VIC-1525, Seikosha GP 100 VC o Commodore MPS 803, tutte perfettamente compatibili tra loro, quindi qualsiasi stampante tra queste si possegga sarà sempre necessario riferirsi alla versione dell'ADP Basic per la 801.

Inserito il dischetto si dovrà semplicemente caricare il programma LOADER e dare RUN: il 64 ci chiederà di digitare 801 e 802 a seconda di quale stampante possediamo (ripetiamo, 802 solo per questa, 801 per tutte le altre...)

Sentiremo in movimento il drive per qualche secondo ancora, e... tutto qui. l'ADP Basic sarà pronto per accettare i nuovi comandi implementati.

Non sono necessarie SYS o operazioni di alcun genere; provate ad esempio a leggere la directory del disco con il comando CAT.

Con questo speriamo di aver semplificato la vita ai più, se avete qualche altro problema o suggerimento, ripetiamo, scriveteci pure.

MC

Osborne è tornata

Pensiamo che come la Osborne pochi nomi nella storia del microcomputer siano stati tanto menzionati dalla stampa sia nel bello che nel cattivo tempo

Giustamente definita come pioniere, per aver introdotto sul mercato un computer portatile e aver dato l'avvio a una drastica riduzione dei prezzi si potrebbe dire che della Osborne se ne e parlato forse in ragione superiore alla sua posizione occupata nel mercato del microcomputer: non superiore al 2% del mercato mondiale Cosa ancora piu strana, ma in questo gli americani sono maestri, se ne e parlato ancora di piu quando gli affari in casa Osborne non andavano per il verso giusto In Italia una situazione di bancarotta sarebbe stata tenuta segreta al massimo, per non nuocere al buon nome a al prestigio acquisiti negli USA invece tale situazione e stata usata come fonte di pubblicita gratuita Si puo dire con quasi certezza che la Osborne e stata menzionata dalla stampa molto di piu quando si trovava sotto amministrazione controllata La stampa italiana comunque non ha seguito questo andamento per analizzare i fatti di casa nostra possiamo dire che della Osborne ce ne siamo occupati sempre con piacere; comunque, se in passato non abbiamo dato molta rilevanza ai fatti Osborne, cio é dovuto alla nostra consuetudine di non dare eccessivo rilievo a fatti inerenti marchi che non sono commercializzati sul nostro territorio dobbiamo ricordare che praticamente la Osborne e da piu di un anno che non e rappresentata in Italia

Orbene, avendo da poco ricevuta la comunicazione che la Osborne e ufficialmente rappresentata in Italia dalla Computator di Roma, siamo lieti di darne la notizia e tracciare a beneficio dei nostri lettori, un sommario degli avvenimenti Osborne, dando particolare rilievo agli ultimi fatti che hanno permesso alla stessa di occupare per la seconda volta una posizione di prestigio nel mercato del microcomputer

Adam Osborne con l idea di produrre microcomputer, nel settembre 1979 formo una societa denominata BrandWine Holdings Ltd e successivamente Osborne Computer Corporation, nel marzo 1980 lo stesso ingaggio Lee Felsenstein, progettista di microelettronica, per il progetto di un microcomputer portatile economico, in versione completa e che potesse trovare posto sotto il sedile di un aeroplano, punto di forza per l'introduzione sul mercato di tale apparecchio fu la strategia di fornire la macchina completa di software applicativo compreso nel prezzo I primi esemplari fecero l'apparizione sul mercato nel giugno 1981 con il software compreso (il cui costo era quasi pari al costo della macchina) e con un prezzo di circa la meta rispetto a configurazioni presenti sul mercato e con caratteristiche equiparabili, l'Osborne I si dimostro un sicuro successo, nei primi otto mesi furono venduti 11 000 esemplari con richieste superiori a 50 000 unita e in circa un anno la casa riuscì a consegnare piu di 10 000 unita al mese

Probabilmente questa rapida crescita porto alla rovina di una iniziativa nata sotto i migliori auspici prestiti bancari, finanziamenti improduttivi, investimenti su futuri profitti, manie di grandezza e spese faraoniche stavano ben coperti dietro la facciata di una crescita rapida e sicura Analizzando questi fatti a posteriori, forse oggi si puo dire che la Osborne non sia stata mai in attivo Nel marzo 1982 venne inaugurata una nuova catena di montaggio nel New Jersey con una capacita produttiva di 200 unita al giorno in vista del lancio del nuovo modello Executive in fase di progettazione A fine 1982 la Osborne contava piu di mille dipendenti Nel contempo il mercato del microcomputer cominciava a dimostrare maggiore interesse per il portatile, se all'inizio il mercato del portatile era esclusivo monopolio Osborne, l'avvento della concorrenza (Kaypro, Compaq, ecc) portava a una riduzione dei margini di profitto Inoltre il quartier generale della compagnia veniva spostato in un palazzo faraonico circa 5 miliardi per fitto e modifiche all'immobile

La necessita di amministrare una cosi grande organizzazione a livello professionale porto alla nomina quale capo dell esecutivo, di Robert Jaunich II che in precedenza era presidente della Consolidated Foods Corporation questa decisione si rivelo fatale per una organizzazione gia piena di problemi come primo risultato venne annunciato il modello Executive a un prezzo non molto competitivo, il progetto del piu piccolo modello Vixen, che avrebbe dovuto sostituire l'Osborne I, venne momentaneamente accantonato e le residue finanze vennero investite su un modello IBM compatibile, per colmare la misura i prodotti annunciati non erano pronti per la produzione e molti mesi sarebbero passati prima della loro introduzione sul mercato, inoltre, quando il modello Executive fu effettivamente disponibile (nella tarda primavera 1983), non fu certo benefico l'annuncio di un successivo modello IBM compatibile come risultato pressoche immediato si fermarono le richieste dei modelli disponibili

Con le entrate ridotte al minimo e spese fisse molto alte si arrivava, nell'agosto 1983 al licenziamento della maggior parte dei dipendenti Le banche cominciavano a rifiutare i crediti e i creditori a richiedere le loro spettanze e quindi il 13/9/1983 la Osborne si trovava costretta a far ricorso alla protezione del cap 11 del codice di bancarotta americano con un deficit di circa 45 milioni di dollari Durante il periodo di amministrazione controllata molti potenziali acquirenti cercavano di mettere le mani sulla Osborne ma a condizioni a dir poco inaccettabili

Nel novembre 1983 lo staff della sezione internazionale si presentava con un piano di riorganizzazione che non veniva accettato, uguale sorte toccava al distributore canadese che aveva proposto di acquistare parte del patrimonio Si era quindi a un passo dalla liquidazione per cercare di evitare la liquidazione ormai certa, la Osborne il 6/12/1983 rinunciava ai quattro mesi di esclusiva a lei spettanti per la presentazione del piano di riorganizzazione e contemporaneamente eleggeva Ronal J Brown quale presidente e capo dell'esecutivo Il 27/1/1984 il suddetto presidente assieme ad altri tre membri dello staff internazionale (Chodi Mc Reynolds, David Miller e James Schwabe) presentava, con l'aiuto di un valido avvocato fallimentare, un nuovo piano di riorganizzazione che alla fine, dopo aver ottenuto il consenso dei creditori, veniva approvato dalla corte il 14/3/1984

Conseguenza primaria dell'applicazione di tal piano di riorganizzazione era la riduzione drastica delle spese (dai tre milioni di dollari mensili a 250 000) e il completamento e l'acquisizione di nuovi prodotti gia in fase di sviluppo

Ci si potrebbe adesso domandare cosa e rimasto della precedente organizzazione forse ben poco, ma tutto quello che e rimasto e certamente

Arision, la casa elettronica secondo Ariston

Ariston, grossa produttrice europea di elettrodomestici, ha presentato a Milano un sistema di gestione dell'abitazione domestica controllato da un home computer tramite Arision, una piccola scatola nera di interfacciamento

Con Arision, realizzato in collaborazione con la societa ISI (Ital Sistemi per l'Informatica) di Roma, si puo controllare il funzionamento di forno, scaldabagno, lavastoviglie, lavabiancheria eccetera, fino alla segreteria telefonica, migliorando le prestazioni e riducendo i costi Inoltre Arision esclude i componenti che non funzionano e tramite Teledrin (un particolare radiotelefono) può chiamare il padrone di casa ed avvertirlo di cosa bolle in pentola!

Il sistema si basa su un microprocessore Motorola 6800, 4K di programma su Eprom e 2K Ram, la comunicazione con il computer avviene tramite interfaccia RS 232C con velocita di trasmissione fissa, pari a 4800 baud All'interno e contenuto un combinatore telefonico per l'uso come segreteria telefonica, e anche un sintetizzatore di voce, utile sia per il messaggio da lasciare in propria assenza che per farsi chiamare in caso di guasti Per l'uso come segreteria serve ovviamente un registratore per tenere i messaggi

È prevista una scheda di riconoscimento della voce

Arision è la più giovane industria europea di elettrodomestici, e si colloca al 5° posto della classifica produttori, dietro a Zanussi-Electrolux, Philips-Bauchnecht, Bosch-Siemens e Thomson

Per ulteriori informazioni.
SCR Associati
Foro Bonaparte 70 - 20121 Milano

Personal Maint: assistenza Apple ed IBM a Roma

Sulla scia di quanto sta avvenendo negli Stati Uniti, anche in Italia c'è chi organizza servizi di manutenzione per personal computer indipendenti da quello del costruttore a Roma e entrata in attivita dal primo marzo la Personal Maint S r l , specializzata nell'assistenza a prodotti Apple ed IBM Alberto Gagliardini e Massimo Momola, fondatori della Personal Maint, vantano una ultradecennale esperienza nel settore delle apparecchiature elettroniche

Per ulteriori informazioni
Personal Maint
Piazza di Villa Carpegna 52 Roma
Tel. (06) 6223990

il meglio e degno di essere salvato, forse sarebbe più facile elencare quello che della vecchia organizzazione e scomparso, assieme al fondatore e al presidente sono scomparsi anche molti debiti dei circa quindici milioni di dollari dovuti ai creditori garantiti ne rimangono da pagare meno di quattro L'attuale societa conta circa trentacinque dipendenti che non diventeranno più di cinquanta nei prossimi due anni

Attualmente la società si trova presente sul mercato con due nuovi modelli in aggiunta ai due precedenti, ma di questo daremo notizia nei prossimi numeri

Per ulteriori informazioni
Computator s.r.l , via F Verdinois 8
00159 Roma Tel (06) 4563008

Melchioni lancia il Toshiba MSX

Sbarca in Italia il Toshiba HX-10, un home computer della lunga fila MSX, standard giunto ai 40 produttori, non solo giapponesi ma anche europei ed americani Le principali caratteristiche di questo computer, come ovviamente di tutti gi MSX, sono il microprocessore Z80 con 64K Ram e 32K Rom, un chip specializzato per il suono (l'8912 della GI) e uno per la grafica (Texas), una tastiera da 73 tasti multifunzione, interfacce per stampante o plotter

Una caratteristica che rende interessante l'HX 10 è il manuale, gia tradotto in italiano Per quel che riguarda il software, garantito dalla compatibilita di tutti gli MSX, la stessa Toshiba ha realizzato i principali programmi applicativi T-Plan , uno spreadsheet, T-Graph, grafica a colori con il plotter; T-Painter, per disegnare sullo schermo tramite il joystick, Bank Street Writer un wordprocessor

Il Toshiba usera i nuovi dischetti da 3,5 pollici da 1 MB Altre periferiche Toshiba sono la stampante HX-P550, a matrice di punti, e il plotter HX-P570, che può anche listare

Per ulteriori informazioni
Melchioni S.p A
Via P. Colletta 37 Milano

Tiber: nuovi personal Toshiba

È stata annunciata in questi giorni la nuova gamma di personal computer della giapponese Toshiba, distribuiti in esclusiva per l'Italia dalla Tiber di Roma Quattro i modelli il T/1500 un compatibile totale PC e XT della IBM, l'XBC, un portatile con incorporato un video a cristalli liquidi, il T/350, compatibile con l'AT della IB infine il T/1500B con microprocessore 8086-2 e hard disk fino a 20 MB Il primo sara in distribuzione da marzo, gli altri modelli seguiranno nel primo semestre del corrente anno Per il 1500 versione base, che è completamente compatibile con il precedente Toshiba T/300, il prezzo e fissato in 4 945 000 lire piu IVA la versione base comprende l'unita centrale, 128K Ram tastiera monitor monocromatico, doppio floppy da 360 × 2 KB, cavi di collegamento, manuali e sistema operativo Un floppy può essere sostituito da un hard disk da 10 MB, per un prezzo complessivo di 7 945 000 lire

Per ulteriori informazioni
Tiber Attrezzature ufficio
Via Madonna del Riposo 127 - 00165 Roma

Memorex protegge i dischetti

La necessità di compattare all'interno di un dischetto la maggior quantita possibile di dati ha portato alle capacità record di 3,3 MB in un unico floppy da 5,25", mentre e normale una capacita di 1,6 MB L'elevata densita di registrazione ha ovviamente messo alla frusta i punti deboli di questo supporto, particolarmente sensibile alle infiltrazioni di polvere e agli stress meccanici Per prevenire i disastrosi effetti di queste cause, gia da tempo la Memorex aveva apportato delle innovazioni nella realizzazione dei suoi dischetti i bordi vengono saldati lungo l'intero perimetro, e non solo per punti come fa la maggior parte della concorrenza, la stessa custodia plastica del floppy viene realizzata in materiale rigido, che assorbe la maggior parte degli urti e delle torsioni

Adesso la Memorex ha completato la sua rete di protezione, aggiungendo un terzo livello I suoi dischetti vengono infatti venduti in confezioni da 10 elementi, contenuti in una scatola di plastica rigida, al costo globale pari a quello di 11 dischetti in questo modo la trasportabilita e garantita, e i dischetti possono ben figurare anche in libreria, grazie all'opportuno contenitore

In questo modo si raggiungono le condizioni ottimali per la salvaguardia del disco, e quindi dei dati che contiene

Per ulteriori informazioni
Memorex Italia
Via C. Menotti 14
20129 Milano

Condor presenta le novità Digital Research

La Condor Informatics di Milano annuncia di aver acquisito la rappresentanza italiana della Digital Research, che allo scorso Smau milanese ha presentato due prodotti destinati ad una vastissima utenza Starlink e Presentation Master. Il primo e un hard-soft che trasforma un PC (modello base o versione XT) oppure un qualsiasi PC compatibile in un sistema multiutente con 4 posti di lavoro, ogni terminale diventa ovviamente un elaboratore a se. Il sistema operativo e il Concurrent PC-DOS. Il prezzo americano è di circa 2000 dollari

Veniamo al secondo prodotto. Si tratta di Presentation Master, il cui scopo ultimo è il trasferimento su diapositiva di tutto cio che appare sullo schermo di un PC (base, XT o compatibile), nel package e compresa una macchina fotografica con obiettivo da 35 mm ed una stampante istantanea per pellicole, in modo da avere la diapositiva nel piu breve tempo possibile. Le immagini sono memorizzate su disco, in modo da rendere eventuali aggiornamenti semplici e veloci. Presentation Master costa 2.500 dollari

Per ulteriori informazioni
Condor Informatics Italia
Via Grancini 8
20145 Milano

Sinclair C5 : la macchina elettrica dello zio Clive

È costruita dalla Sinclair (Vehicles, e non Research), costa quanto un QL, viene consegnata entro 28 giorni (risate...), ma non è un computer. Che cos'è? È la macchina elettrica Sinclair, l'ultima trovata dell'ineffabile zio Clive!

Se ne parlava già da parecchio tempo e finalmente è arrivata, suscitando, come ormai tutti gli oggetti targati Sinclair, molta curiosità ed un'ondata di critiche.

Più che altro si tratta di un triciclo dotato di una carrozzeria in polipropilene e mosso da un motorino elettrico alimentato a batteria, costruito dalla ditta italiana Polymotor. La velocità massima è di 30 Km/h e l'autonomia di una trentina di chilometri. Non molti vero? Ma diventano subito 60 se ci si porta dietro una batteria di riserva ed in caso di emergenza rimangono sempre i pedali, gentilmente forniti di serie

La C5 non ha un volante, ma al suo posto usa un manubrio tipo motocicletta, situato sotto le gambe del pilota, sul quale si trovano tutti i comandi necessari alla guida.

Sentite un po' gli accessori ordinabili a parte specchietti retrovisori, indicatori di direzione, clackson e tutta una serie di impermeabili per proteggere macchina e autista nei giorni di pioggia (tanto in Inghilterra sono rari...)

Non ci vuole la patente per guidarla, basta avere almeno 14 anni

Oltremanica, pero, la C5 ha provocato non poco scompiglio appare infatti minuscola e instabile, e sono in molti coloro che gia si immaginano decine di giovanissimi schiacciati da TIR con rimorchio mentre vanno tranquilli per strada con la loro vetturetta elettrica

Abbiamo visto un esemplare di C5 esposto alla 15 esima ZX microfair, svoltasi il 9 febbraio in una Londra innevata e stretta in una morsa di gelo polare, e vi possiamo assicurare che nessuna ricompensa, per quanto allettante, potrebbe convincerci ad andare in giro per Roma su un aggeggio del genere

FACE: mito e realtà della comunicazione elettronica

La rapida trasformazione tecnologica e la sua evoluzione, nonché le conseguenti modificazioni economiche, e quindi sociali, sono il tema portante del convegno tavola rotonda organizzato dal gruppo FACE, consociato della ITT, lunedì 18 febbraio in occasione del Salone dell'Informatica e dell'Organizzazione Aziendale (SIOA) di Bologna

Ricordiamo che il gruppo FACE (fondato nel 1909), composto oggi da 17 società a capo delle quali troviamo la FACE finanziaria, oltre a vantare 14 000 dipendenti e 800 miliardi di fatturato nell'83, ha operato, sin dall'inizio della sua attività, nel settore delle telecomunicazioni, e non solo, apportando un contributo significativo allo sviluppo della rete telefonica nazionale

La tavola rotonda, che propone il titolo "La comunicazione elettronica mito e realtà", è stata introdotta da Umberto Ferroni, vicepresidente esecutivo ed amministratore delegato della FACE Finanziaria, e coordinata da Franco Morganti di Reseau

La problematica fondamentale, conduttrice del dibattito, è quella che nasce alla luce della seguente domanda in che misura siamo preparati a recepire, umanamente ed economicamente, i cambiamenti introdotti dalla profonda trasformazione tecnologica apportata dalla telematica?

In questo settore sorge una vasta fascia di applicazioni centrate sullo scambio delle informazioni Gli economisti trattano ancora con una certa difficoltà questa nuova "merce", l'informazione, che crea quindi un panorama di incertezze — non inconsueto nella storia — dovuto alla naturale fase di transizione tra le vecchie tecnologie e quelle più rivoluzionarie Per di più, una vastissima fascia di cittadini, di potenziale utenza, non è effettivamente al corrente delle profonde modificazioni in atto ed è proprio questa situazione che li sottopone al rischio di subire le nuove tecnologie piuttosto che gestirle rendendosi conto degli effettivi vantaggi da esse introdotti

Questa situazione è naturalmente indotta dalla scarsa informazione, a cui la scuola soppenisce con fatica proponendo alla società cittadini culturalmente non preparati ai nuovi sviluppi tecnologici, e dalla macchina governativa, come sempre, un'inerzia non irrilevante In pratica, ciò che si chiede è una maggiore consapevolezza e dinamismo da parte degli enti pubblici che, insieme alla tecnologia ed alla massificazione culturale, produrrebbe, in termini economici, un aumento della domanda

Alla base della società telematica c'è la necessità di una integrazione operativa, spinta al limite della unificazione delle reti già esistenti, che viene sintetizzata dall'unificazione dei protocolli, e delle modalità di comunicazione in generale, che permetterà un più agevole scambio dell'informazione

Lo sviluppo delle telecomunicazioni infatti, per ovvi motivi, è avvenuto in maniera abbastanza disorganica

Ogni tipo di informazione da trasmettere si serve infatti di un mezzo diverso telegrafo e telex per i messaggi, televisione per immagini in movimento, reti computerizzate per l'accesso alle banche dati, ecc Il mezzo che più si presta a soddisfare il desiderio di standardizzazione del sistema mondiale di comunicazione, visto il gran numero di apparecchi terminali oggi in funzione, è la linea telefonica opportunamente ottimizzata

Le linee attuali, predisposte per comunicazioni analogiche (voce) con banda intorno ai 3 kHz, presentano caratteristiche che si adattano poco alle esigenze delle comunicazioni, trattanti informazioni in maniera numerica, a causa della scarsa affidabilità e della lentezza di transito del dato.

Il primo passo verso l'ottimizzazione acquisterà una propria realtà con la creazione di una rete telefonica digitale, l'IDN o Integrated Digital Network, costituita da centrali di commutazione elettroniche che potranno comunicare tra loro in tecnica digitale, mentre il collegamento con l'utente rimarrà di tipo analogico Il passo successivo sarà quello della digitalizzazione fino al livello del terminale d'utente con ISDN o Rete Digitale Integrata nei Servizi, che permetterà collegamenti ad alta velocità pur utilizzando l'attuale "doppino" telefonico che integrerà tutti i tipi di dati da trasmettere Un salto di livello notevole sarà invece apportato dalle fibre ottiche (la Sirti, una delle società consociate, ha posato nell'84 1000 km di cavo e ne poserà altri 2500 nell'85) con le quali sarà resa agevole, come evento culminante, la trasmissione di immagini in movimento

Un esempio della versatilità, semplicità d'uso e affidabilità la troviamo allo stand FACE presso il quale è stata presentata la centrale 1240, di cui sono installati già 11 esemplari, che permetterà lo scambio simultaneo di più forme di informazioni ottimizzando l'uso del doppino telefonico e la sua flessibilità le permetterà di crescere modularmente per soddisfare sia l'aumento dei servizi richiesti che quello degli utenti

La produzione del software applicativo: evoluzione e prospettive

di T. Bettinazzi - A. Maserati
F. Monzani - E. Spoletini - E. Toscani
a cura di E. Spoletini
Collana dei Quaderni di Informatica
Franco Angeli Editore
214 pagine, cm 14 × 22 circa
1983, Milano lire 18.000 IVA inclusa

Questo volume appartiene, come altri già recensiti in passato, ad una interessante collana edita da Franco Angeli, che prende il nome dalla rivista "Quaderni di Informatica", nata nel 1974 per iniziativa della Honeywell Information Systems Italia. La collana comprende una decina di volumi, di vari autori, dedicati ad aspetti tra i più vari dell'informatica, una disciplina giovane ed in rapida evoluzione (linguaggi, evoluzione dell'hardware, elaborazione dati, gestione aziendale).

Il libro è scritto da un gruppo di consulenti e docenti legati in vario modo alla Honeywell. Strutturato in sei capitoli e tre appendici, offre una panoramica della evoluzione e delle prospettive riguardanti la produzione di software applicativo per sistemi di dimensioni medio-grandi. Il lettore che conosce solo i personal o che si limita a scrivere semplici programmi in Basic come quelli che vengono normalmente presentati sulle pagine di MC, viene introdotto alle problematiche, relativamente più ovvie per gli addetti ai lavori, che hanno accompagnato, nel corso degli anni '70, la trasformazione della produzione del software da fatto artigianale a disciplina ingegneristica. Aggiungiamo che gli autori fanno proprio il concetto più ampio di "programmazione strutturata" intendendo con la dizione proprio la produzione del software utilizzando metodologie di tipo ingegneristico, un concetto particolarmente interessante ed ancora, purtroppo, poco diffuso.

Dopo due capitoli a carattere introduttivo, destinati alla definizione di software ed alla evoluzione dei sistemi di elaborazione dati dagli anni '50 agli anni '70, il terzo capitolo fa il punto su quello che gli autori definiscono "il '68 dell'informatica", scatenato da un articolo pubblicato sul n° 4/1968 di *Communications of ACM*, da Dijkstra, un teorico olandese impegnato in attività di consulenza. In sostanza, l'anatema lanciato verso il software che contenesse lo statement GOTO mise in moto una serie di azioni che portarono alla nascita delle moderne metodologie di sviluppo del software, dei tool, dei linguaggi cosiddetti strutturati, alla crescita di importanza del controllo delle interfacce, della documentazione, della analisi sulla fase di codifica vera e propria. Il quarto capitolo presenta un altro componente essenziale del software moderno, il cosiddetto *ciclo di vita*, cioè l'insieme delle fasi lungo le quali si sviluppa un progetto: analisi dei bisogni utente, definizione dei requisiti, progetto di massima, progetto di dettaglio, programmazione, verifica, uso e manutenzione. Viene fatto notare come il costo di un errore, cioè del classico *bug*, sia tanto maggiore quanto più tardi viene scoperto e come ciascuna attività debba essere conclusa da una revisione critica del lavoro svolto prima di passare a quella successiva. I restanti due capitoli e le appendici descrivono in un certo dettaglio alcuni strumenti, metodologie o semplici approcci culturali alla programmazione strutturata: il progetto top-down o bottom-up, il metodo LCP, il metodo Jsp, o quello Pascal-like, nonché la metodologia Phos, messa a punto dagli autori e particolarmente utile per la programmazione in ambienti Data Base o Tele Processing.

Una buona bibliografia completa il lavoro fornendo sia i riferimenti citati, che offrendo, al

lettore meno esperto, ottime basi di partenza per l approfondimento

Scritto con pecisione, ricco di informazioni e di contenuto questo volume risulta interessante ed importante sia per colui che si avvicina professionalmente al software, che per i lettori più attenti e preparati i quali vi trovano una fotografia di come si produce o troppo spesso si dovrebbe produrre il software nei moderni ambienti industriali

Alberto Morando

Chiavi per lo ZX Spectrum

J F Sehan
Edizioni E P.S.I Milano
Distribuzione ETMI
Via Basilicata
20098 S Giuliano Milanese (MI)
112 pagine - 15 × 21 cm - Lire 11.000

Questo libro contiane una raccolta di tutte le informazioni necessarie al programmatore dello Spectrum, a partire dall'elenco dei comandi del Basic per arrivare alla succinta descrizione delle principali routine della ROM

Si tratta, in pratica, di un prezioso vademecum da tenere a fianco del computer per poterlo consultare quando serve

L'idea di raccogliere in un unico volumetto tutti i fatti fondamentali relativi allo Spectrum, per evitare a chi ricerca un dato di perdersi in troppi testi diversi, è senza dubbio ottima, e questo libricino può rivelarsi un amico indispensabile in un gran numero di occasioni

Data la particolarità dell' opera e stata adottata una veste grafica estremamente scarna, volta a non distrarre l'occhio del lettore su particolari inessenziali Sul lato superiore destro di ogni pagina e presente una striscia colorata, all'interno della quale si trova il nome della sezione cui la pagina appartiene Questo permette di aumentare ancora la velocita di ricerca del punto di proprio interesse basta infatti sfogliare rapidamente il volume prestando attenzione alle scritte sopra citate per identificare subito la sezione desiderata

Gli argomenti dell'opera sono, nell'ordine comandi del Basic e loro sintassi, codici di errore, set di caratteri ed organizzazione dello schermo, istruzioni dello Z 80, piedinatura dei connettori, routine della ROM e variabili di sistema Concludono il volume una serie di utili trucchetti ed un comodo indice analitico

Maurizio Bergami

Progetti hardware con lo ZX Spectrum

G Bishop
McGraw-Hill Book Co GmbH
Lademannbogen 136
2000 Hamburg 63, RTF
176 pagine - Lire 17 000

Non esiste possessore di personal o home computer che non sia affascinato dalla possibilita di interfacciare il suo computer al mondo esterno Purtroppo non sono molti gli appassionati di microinformatica che hanno le conoscenze di elettronica necessarie per tentare qualche piccolo esperimento nel campo

Ci vuole quindi l'aiuto di qualcuno, e per gli utenti Spectrum l'aiuto arriva sotto forma di questo libro, edito dalla Mc Graw-Hill e scritto da Graham Bishop

L'opera e impostata essenzialmente su un piano pratico dopo i primi due capitoli dedicati ad una breve analisi delle caratteristiche hardware e software dello Spectrum, l'autore parte con i primi progettini

Le proposte presentate sono troppo numerose per essere citate tutte, tra quelle più interessanti troviamo un convertitore analogico-digitale ed uno digitale-analogico, una penna ottica ed un sistema antifurto

Per non ridurre tutto ad un mero assemblaggio di alcuni componenti elettronici, Bishop si preoccupa di dare il giusto spazio alla teoria, con il duplice scopo di fornire gli elementi necessari alla comprensione del funzionamento del circuito e le basi per tentare eventuali modifiche personali

Per ogni progetto viene fornito il necessario

supporto software, ed inoltre un'apposita appendice riporta il disegno di tutti i circuiti stampati

È bene avvertire che non si tratta di un libro diretto ai principianti assoluti il lettore ideale dovrebbe già avere una pratica, anche se minima, di montaggi elettronici e, possibilmente, conoscere i rudimenti del linguaggio macchina Z80

La veste grafica, la traduzione e la realizzazione sono di livello qualitativo molto elevato, come è costume della Mc Graw-Hill Il prezzo, infine, è perfettamente in linea con l'ottimo contenuto del libro

Maurizio Bergami

32 programmi con l'Apple

di Tom Rugg e Phil Feldman
Franco Muzzio & C. Editore
Via Makallè, 73 - 35138 Padova
Padova - 1981
252 pagine, 12.000 lire

Una raccolta di programmi di tutti i generi e di tutti i tipi giochi, utilità, matematica e applicazioni domestiche Il classico libro per chi, avendo appena acquistato la macchina, e non essendo quindi ancora in grado di sviluppare le proprie applicazioni, non voglia restare con il computer inattivo In questo libro, al solo costo della fatica di copiatura e debug, puoi trovare tutti quei programmi da far vedere a chi, alla vista del vostro gioiello, pone la classica domanda "Bello, ma cosa sa fare?"

Tutti i programmi sono commentati e accompagnati da fotografie delle schermate, inoltre vengono consigliate le modifiche da apportare per migliorare ulteriormente alcuni punti o personalizzare certe uscite Molto estesi gli esempi matematici con programmi di soluzione per equazioni differenziali o il calcolo del polinomio interpolante con il metodo dei minimi quadrati Tanti pure i giochi e le applicazioni per la casa, dal bilancio domestico al calcolo degli interessi su un prestito, a vari programmi educativi sia di aritmetica che di italiano o inglese

Al costo del libro, non eccessivo se si considera il tempo di sviluppo dei programmi, bisogna aggiungere il tempo necessario alla copiatura (alcuni programmi sono molto lunghi), ma questo è comunque un esercizio utile per prendere confidenza con la tastiera dell'Apple che, almeno nel IIe, è particolarmente complicata.

Valter Di Dio

Machine Code Graphics and Sound for the Commodore 64

di M. England e D. Lawrence
Ed. Sunshine, 12-13 Little Newport ST London WC2R 3LD
pag. 216 cm 23 × 15, £6 95

David Lawrence è la punta di diamante della Sunshine, per cui lavora sia come autore di libri (The Working Commodore 64 e Advanced Programming Techniques per il 64) che come articolista sui vari periodici della casa. Mark England è stato lanciato dalla coproduzione, insieme a Lawrence, del libro C64 Machine Code Master recensito su MC 30, sull'onda di quel libro, è stato prodotto come software l'assemblatore-disassemblatore Mastercode, ed è stata lanciata una nuova maniera di far libri. Infatti anche quest'opera, come altre della stessa Sunshine, offre non solo routine in linguaggio macchina, bensì istruzioni che possono essere aggiunte al Basic del vostro 64 nel modo indicato tramite il programma Basic Extender.

I 5 capitoli sul LM riguardano utility, bassa ed alta risoluzione, sprite e suono, e per ognuno viene riassunta la teoria di base che c'è dietro l'organizzazione di quel settore del 64: tutte le routine sono disassemblate e commentate in modo più che esauriente, e tutti i listati Basic sono commentati linea per linea. Al termine viene presentato l'elenco dei valori esadecimali necessari a fruire della routine senza dover usare un assemblatore, cosa questa resa possibile dal programma Hex Loader, listato nell'appendice A. Un appunto, peraltro lieve, è che tutti i disassemblati sono basati sul citato Mastercode: che è piuttosto completo (prevede PRT, ORG, SYM, variabili, etichette ed altro), per cui chi non lo compra da loro deve farsi un piccolo lavoro di riconversione su ogni routine.

Le routine possono essere facilmente seguite da chiunque abbia familiarità con il linguaggio macchina del 6502 e con la struttura interna del 64: si tratta in definitiva di un lavoro certamente non adatto ai principianti, ma utilissimo a chi voglia migliorare il grado di sfruttamento del 64 e le proprie conoscenze sulla programmazione in assemblativo.

Il testo è in inglese.

Leo Sorge

Esercizi per l'Apple II, II+, IIe, IIc

di Frèderic Lèvy
Edizioni E P S I
Distribuzione ETMI
Via Basilicata
20098 S. Giuliano Milanese
Milano - 1985
146 pagine 14.000 lire

Finalmente un libro di esercizi nel vero senso della parola, non programmi commentati, ma un vero e proprio libro "scolastico" con presentazione del problema consigli sullo sviluppo del programma e quindi il flow-chart. Nella seconda metà del libro si trovano le soluzioni degli esercizi in Basic.

Naturalmente di Basic non si parla, dando per scontato che chi si accosta ad un libro di esercizi conosca già la grammatica del Basic, mentre, quando serve, vengono rispolverate vecchie conoscenze matematiche come ad esempio l'equazione della circonferenza o il calcolo dei numeri primi.

Il testo è ridotto al minimo indispensabile mentre molto spazio è destinato ai diagrammi di flusso e alle figure esplicative. Molto comode, nelle soluzioni, le evidenziazioni dei cicli principali del programma e la spiegazione sull'uso delle variabili.

Un libro perciò utilissimo per quanti vogliono imparare a programmare e soprattutto a fare della buona analisi. *Valter Di Dio*

e c'è questo simbolo
rsonal computer: Commodore.

per il tuo computer: S.H.R. conosce meglio di chiunque il software Commodore, perche realizza tutti i programmi Commodore da molto più tempo degli altri. E li aggiorna costantemente, per rispondere sempre meglio alle esigenze degli utenti.

S.H.R., cioè hardware e software Commodore con più esperienza e un vantaggio sostanziale: da noi i prodotti costano meno; e quando S.H.R. dice "costa meno", costa meno davvero. Chiedi ai Professionisti dell'Informatica. S.H.R. A conoscerla c'e da guadagnarci.

SHR

Software, Hardware, Ricerca

S.H.R. s.r.l. - Casella Postale 275 - 48100 Ravenna
Tel. 0544/463200

NOVITA' EPSI
Supergiochi: il Commodore 64
30 Programmi in Basic
Autore Jean François Sehan
Pagine 134 Formato 17×24
Brossura cucita ISBN 88 7688 601 X
Lit 18 000
Questo libro si rivolge a coloro che possiedono un Commodore 64 L'interesse immediato è costituito dai trenta programmi di giochi utilizzabili subito tali e quali Essi si possono classificare in giochi di abilità, di riflessione e di fortuna L'altro aspetto interessante e quello pedagogico il libro è utile a chiunque voglia migliorare la propria tecnica di programmazione Infatti per ogni gioco si trova oltre al listato propriamente detto la descrizione del gioco l'analisi sotto forma di diagramma di flusso, il commento delle istruzioni più importanti, una tabella di identificazione delle variabili, per facilitare ogni ricerca ulteriore
La scoperta del PC 1500
1 Programmazione in Basic
Autore Jean Pierre Richard
Pagine 240 Formato 15×21
Brossura cucita ISBN 88 7688 214 6
Lit. 22 000
Il libro si rivolge ai possessori del computer tascabile PC 1500 che intendono imparare il Basic Descritto il ruolo dei nove indicatori, si digitano le prime linee di programma e si imparano le prime nozioni di programmazione Si passa poi alle operazioni aritmetiche, alle variabili numeriche e alfanumeriche e alle nozioni di ortografia, di memoria e di ingresso dati Si introducono i diagrammi di flusso e la messa a punto e ottimizzazione dei programmi A questo punto si possono già scrivere dei semplici programmi Si passa quindi a calcoli e formati più complessi e alle nuove nozioni di loop, trattamento delle stringhe, sottoprogrammi e ordinamento Si conclude con le operazioni sulle stringhe alfanumeriche e con i grafici e certe funzioni particolari
102 Programmi per il Philips C 7420 Videopac +
Autori J Deconchat, V Grandis
Pagine 240 circa Formato 17×24
Brossura cucita ISBN 88 7688 102 6
Lit 21 000
Imparare divertendosi è l obiettivo di questo libro Tutti i programmi sebbene completi e di buone prestazioni sono molto corti Questo facilita l introduzione nella macchina e l apprendimento progressivo del Basic Grazie ai cinque livelli previsti le istruzioni Basic sono presentate per difficoltà crescente come in un manuale introduttivo i giochi come tali sono utilizzabili senza alcuna conoscenza di programmazione Ogni programma è seguito da suggerimenti che propongono modifiche e perfezionamenti
Ogni gioco e così presentato breve scheda segnaletica (categoria e livello di difficoltà, descrizione del gioco, studio dettagliato delle linee importanti di programma, modo di giocare, estensioni possibili
BEST SELLERS
Il Basic illustrato
Un linguaggio semplice di programmazione
Autore Donald Alcock
Pagine 144 Formato 13×21
Legatura ad anelli ISBN 88 214 0593 X
Masson Italia Editori
Lit. 15 000
Una caratteristica singolare di questo libro è il modo in cui si presenta e interamente scritto a mano e illustrato come un fumetto Questo unito a un informazione puntuale, fanno di questo piccolo manuale un libro vincente dal punto di vista didattico per coloro che posseggono un piccolo personal
Particolare attenzione è posta nel descrivere un Basic il più indipendente possibile dalla macchina
A tale scopo sono state accuratamente studiate 11 diverse versioni del linguaggio Basic
Dizionario del Basic
Enciclopedia del linguaggio Basic
Autore David A Lien
Pagine 458 Formato 17×24
Brossura cucita ISBN 88 7688 001 1
Lit 40 000
L'autore di questo volume si propone di riunire la più completa raccolta di parole Basic e di descrivere la strategia che i programmatori possono usare per passare da un "dialetto" all altro Il risultato e un manuale di lavoro che aumenterà moltissimo la vostra capacità di programmare Questa seconda edizione inglese che e stata tradotta in italiano contiene quasi cinquecento parole e in realtà comprende ogni parola significativa usata dai calcolatori che parlano il Basic distribuiti in tutto il mondo Ma l'autore va ancora oltre e attacca il problema della incompatibilità indicando le strategie per convertire i programmi da un calcolatore all altro Sia che voi abbiate un piccolo computer tascabile sia un sistema dell ordine di megabyte questo libro sarà per voi un aiuto prezioso

NOVITA' MASSON
I microprocessori della famiglia iAPX 86
Hardware software sistema operativo
Autore A.B Fontaine
Pagine 230 Formato 17×24
Brossura cucita ISBN 88 214 0567 2
Lit 24 000
Questo fornisce un'esauriente descrizione dell'hardware e del software dei processori elencati e può costituire quindi un ottimo strumento di valutazione degli stessi, nonché uno strumento di lavoro da intendersi come propedeutico rispetto ai manuali piu completi forniti dalla casa produttrice La descrizione dell'architettura dei diversi elementi e l'analisi delle soluzioni hardware, delle metodologie di sviluppo del software, dei linguaggi di programmazione adottabili, nonche dei problemi relativi alle applicazioni in tempo reale e all'impiego di un sistema operativo, forniscono un compendio che può essere di valido aiuto indipendentemente dai riferimenti alla particolare famiglia
Fortran 77
Il linguaggio Fortran V
Autore Patrice Lignelet
Pagine 190 circa Formato 17×24
Brossura cucita ISBN 88 214 0571 0
Lit 18 000
Questo libro e una guida dettagliata al Fortran V ed è corredata da numerosi esempi che consentono anche al lettore meno esperto di apprenderne a fondo le caratteristiche per questo essa si rivolge sia a chi già conosce il Fortran sia a chi si avvicina per la prima volta Una visione ancora piu completa si avra leggendo il corrispondente volume di esercizi di prossima pubblicazione Contenuti Elementi di base del linguaggio I dati I primi passi leggere scrivere calcolare. istruzioni condizionali I cicli Le procedure sottoprogrammi e funzioni Le tabelle Gli archivi sequenziali e ad accesso diretto Input/output formattati Il volume si rivolge a tutti coloro che desiderano apprendere il linguaggio Fortran V
Programmare M 68000
Autore Tim King, Brian Knight
Pagine 180 circa Formato 17×24
Brossura cucita ISBN 88 214 0572 9
Lit 18 000
Gli autori introducono le idee base dell'architettura di M 68000 e spiegano le caratteristiche di ogni istruzione compresi i trabocchetti e gli impieghi particolari Vengono forniti molti esempi pratici, gli strumenti per il debugging, le routine di interrupt e i programmi per l'allocazione di aree di memoria Tutti gli esempi sono stati provati, sono ben commentati e descritti e possono essere usati come frammenti per programmi piu complessi Una caratteristica particolare è un piccolo programma monitor completo che gestisce ingresso e uscita e permette ai programmi di prova di girare Il libro sarà uno strumento prezioso sia per appassionati in genere sia per i tecnici che impiegano questo particolare microprocessore Non si richiede pratica di programmazione in assembler, anche se il lettore deve conoscere la terminologia informatica elementare
Metodo generale di analisi di un'applicazione informatica
vol 1 Fasi e punti fondamentali dell'analisi funzionale
Autore Xavier Castellani
Pagine 310 circa Formato 17×24
Brossura cucita ISBN 88 214 0569 9
Lit 25 000
L opera illustra come si conduce l'analisi di un'applicazione informatica di tipo gestionale (automatizzazione di procedure di paghe e stipendi, gestione magazzino ecc) Questo volume si concentra in particolare sulla fase di studio delle procedure che si desidera informatizzare e di messa a punto delle soluzioni hardware, software e organizzative da adottare Tale fase è detta fase di analisi dell applicazione L'opera si rivolge ai tecnici informatici (capi progetto, analisti, analisti programmatori, programmatori) agli utenti studenti e docenti di informatica Gli argomenti sono trattati in modo molto puntuale e dettagliato, pertanto il libro può essere letto oppure consultato come un manuale È ricco di esempi ed alcuni punti a contenuto particolarmente tecnico sono illustrati con l'aiuto degli allegati
COBOL vol 1
La logica della programmazione
Autore Michel Koutchouk
Pagine 160 circa Formato 17×24
Brossura cucita ISBN 88 214 0570 2
Lit 20 000
L'opera e un corso completo di programmazione in COBOL. Questo volume si rivolge al principiante e permette di acquisire in modo graduale una conoscenza completa del linguaggio e un metodo di costruzione dei programmi (il metodo di costruzione logica dei programmi di Warnier) A questo proposito l autore insiste su due punti fondamentali un programma deve essere modificabile e portatile, deve di conseguenza essere strutturato con l'aiuto di un ragionamento logico semplice e accessibile a tutti ciò a causa del carattere mutevole dei problemi dell'impresa e della necessità di trasferire i programmi da una macchina all altra e anche da un codice di programmazione ad un altro
La matematica e li personal computer
Autori H Lehning, D Jakubowicz
Pagine 456 Formato 17×24
Brossura cucita ISBN 88 214 0566 4
Lit 11 000
Argomento tra le discipline che maggiormente possono beneficiare di un rinnovamento didattico, grazie all'introduzione dell elaboratore, vi è la matematica che spesso si mostra ostica allo studente per la difficolta della comprensione del suo linguaggio formale Il libro affronta alcuni semplici argomenti della matematica con l'ausilio dei concetti basilari dell'informatica Particolare attenzione viene posta alla nozione di algoritmo mentre vengono introdotti i primi elementi dei linguaggi BASIC.

La pratica dello ZX Spectrum

vol 1 Basic approfondito e introduzione al linguaggio macchina
Autore Xavier Linant de Bellefonds
Pagine 160 Formato 15×21
Brossura cucita ISBN 88 7688 **202** 2
Lit 15 000

Destinato a completare la documentazione di base dello ZX Spectrum questo manuale sfrutta tutte le possibilità del Sinclair nell'ambito della programmazione avanzata
L autore vi espone in maniera accuratamente graduata la definizione di caratteri e le funzioni di stringa (archivi alfabeti trattamento di testi) l'alta definizione grafica e la potenza di calcolo (regressione, grafici a tre dimensioni) il colore e il suono (svariati giochi), le tecniche di accesso diretto alla memoria e di utilizzo del linguaggio macchina
I possessori dello ZX-81 troveranno in questo manuale numerosi programmi trasferibili che possono essere introdotti tali e quali sul loro calcolatore

La pratica dello ZX Spectrum

vol 2 Programmazione in linguaggio macchina
Autore Marcel Henrot
Pagine 164 Formato 15×21
Brossura cucita ISBN 88 7688 **203** 0
Lit 15 000

Quest opera dedicata alla programmazione in linguaggio macchina è accessibile a tutti coloro che abbiano assimilato la programmazione in Basic avanzato
La correlazione tra i comandi Basic ed i codici macchina è stata la preoccupazione principale dell autore che porta progressivamente il lettore alla comprensione di routine sempre piu complesse
I tre aspetti principali dello Spectrum il suono, il colore, l'alta risoluzione sono oggetto di numerosi programmi in tutti i capitoli
A poco a poco si analizzano le operazioni logiche, il trattamento delle iterazioni, l'assegnazione particolare dello Spectrum e l'animazione sullo schermo
Il lettore attento quando avrà chiuso questo libro elaborerà dei programmi personali in codice macchina

L'Apple e i suoi files

I Metodi pratici
Autore Jacques Boisgontier
Pagine 180 Formato 15×21
Brossura cucita ISBN 88 7688 **204** 9
Lit 19 000

Per non entrare troppo rapidamente nella programmazione delle applicazioni che utilizzano i files l'opera inizia con una presentazione concisa e illustrata dei comandi del sistema operativo disco e delle istruzioni del Basic Applesoft In seguito vengono descritte le istruzioni dei files sequenziali e ad accesso casuale e viene spiegato il loro impiego con l aiuto di programmi classici di creazione, modifica e ordinamento di files abbondantemente commentati
Alcuni metodi pratici, spesso poco conosciuti, mostrano come utilizzare al meglio i files ad accesso casuale, l'accesso indicizzato l allocazione dinamica, il codice ASCII, le liste inverse
Una ventina di programmi illustrano l'impiego di queste tecniche

Il Basic per tutti

iniziazione piu programmi
Autori Jacques Boisgontier Sophie Brebion
Pagine 160 Formato 17×24
Brossura cucita ISBN 88 7688 **501** 3
Lit 18 000

Imparate a programmare con il "Il Basic per tutti" in mano Mettetevi davanti al vostro computer (Apple Commodore 64 TRS 80) e incominciate a scrivere alcune istruzioni Molto rapidamente assimilerete le nozioni fondamentali della programmazione variabili, test e loop
Riuscirete a "comprendere" il computer e a sapere ciò che ne potrete ricavare Ma non è tutto Grazie ai numerosi esempi illustrati e ai programmi commentati acquisirete senza difficoltà le basi necessarie ad approfondire le vostre conoscenze e a scrivere i vostri programmi di gestione didattici e di giochi

La pratica dell'Apple II

1 Periferiche e gestione dei file
Autore Nicole Breaud Pouliquen
Pagine 116 Formato 15×21
Brossura cucita ISBN 88 7688 **211** 1
Lit 12 000

Questo libro "La pratica dell Apple II" si rivolge a tutti gli appassionati dell Apple È stato scritto soprattutto alla luce delle potenzialità dell Apple IIe
Vi si affronta lo studio dell unità a dischi sia dal punto di vista hardware sia da quello software Diversi capitoli sono dedicati al sistema operativo DOS e alla gestione dei file Vengono studiati i diversi modelli di stampati e le diverse modalità di stampa L'ultimo capitolo è dedicato alla scheda Apple-Clock

Modelli di espressione grafica

Autore Jean Pierre Blanger
Pagine 232 ca Formato 17×24
Brossura cucita ISBN 88 7688 **005** 4
Lit 20 000

Questo testo presenta un insieme di tecniche che illustrano le possibilità delle realizzazioni grafiche del calcolatore
Il suo procedimento progressivo permette al dilettante come al professionista di affrontare la risoluzione di problemi sempre piu complessi (Tracciati d ellisse, rotazione dei poligoni, tratteggio delle superfici) I modelli di espressione grafica presentati sono scritti in Basic Applesoft I programmi, ampiamente commentati, sono facilmente adattabili agli altri calcolatori Ciascun modello e motivo di un analisi, di uno o piu programmi ed esempi di esecuzione che permettono di misurare l'ampiezza delle sue applicazioni

Siel interfaccia Midi per Commodore 64 e Sinclair Spectrum

di Corrado Giustozzi

Di musica e strumenti musicali su MC si è parlato solo in rare occasioni, perché ovviamente, le competenze istituzionali della nostra testata non prevedono interventi di questo tipo. Se però ci capita di mezzo il computer allora le cose cambiano (vedi l'incontro con Doris Norton sul n 32). Questo mese quindi parliamo di oggetti musicali, o meglio, informatico-musicali, presentandovi la prova di un interfaccia che permette il collegamento di uno o più sintetizzatori elettronici ad un C64 o ad uno Spectrum. L'interfaccia colloquia con gli strumenti tramite un protocollo di comunicazione standard sviluppato espressamente per l'interconnessione di strumenti elettronici e denominato MIDI (Musical Instruments Digital Interface). Prodotta dalla italiana SIEL, costruttrice di apprezzate tastiere elettroniche questa interfaccia MIDI è corredata di un opportuno software applicativo che permette anche di comporre musica, archiviare i brani composti e riascoltarli, memorizzare e programmare i timbri degli strumenti ed altre cose ancora. Noi l'abbiamo provata in unione al sintetizzatore Opera 6 della stessa SIEL, una tastiera polifonica programmabile dotata appunto di interfaccia in protocollo MIDI.

Anche se non ce ne sarebbe bisogno specifichiamo che il nostro discorso sarà incentrato soprattutto sugli aspetti tecnici del complesso hard/soft, e quindi ne analizzerò soprattutto le caratteristiche informatiche ed elettroniche. I risvolti maggiormente operativi e musicali/musicologici del collegamento computer-tastiera tramite MIDI, che vanno piuttosto al di là degli scopi di questa rivista, vengono invece trattati già da diversi mesi, dalla nostra consorella AUDIO REVIEW. Nelle sue pagine potrete trovare, oltre ad una serie di articoli dedicati alle problematiche della musica elettronica e dell'informatica musicale in generale, notizie più dettagliate sul protocollo e sulle apparecchiature MIDI. Tra l'altro sul numero in corso viene presentato in dettaglio il MidiScript 64, un potente Editor musicale e Sequencer per il Commodore 64; si tratta di un progetto originale, adatto in particolare all'hardware SIEL. Presentato quindi rapidamente il background di questo articolo, vi invitiamo a proseguire la lettura: l'argomento è forse inusuale, ma decisamente molto interessante.

Generalità sul MIDI

Un sintetizzatore elettronico (vedi anche riquadro a pag. 30), è uno strumento

musicale costituito da un insieme di oscillatori, filtri e amplificatori controllati in tensione. Generando e dosando opportunamente le varie tensioni di controllo, si possono creare suoni aventi un timbro desiderato, con i quali si può suonare una melodia. Il controllo sul timbro lo si ha inviando opportune tensioni ai circuiti di generazione e/o di filtraggio. In questo modo si può facilmente realizzare un controllo esterno automatico di tutte le funzioni dello strumento. Agli inizi della musica elettronica ciò veniva fatto in modo analogico, inviando ai circuiti interessati le tensioni generate da apposite macchine (sequencer), ma con lo sviluppo delle tecniche digitali e dei microprocessori a basso costo ci si è orientati sempre più verso un tipo di controllo digitale/numerico, sia per la generazione del timbro che per il controllo della melodia suonata. Alcuni produttori di tastiere svilupparono pertanto sistemi digitali che permettessero di collegare tra loro più sintetizzatori in modo da poter controllare i parametri di tutti (slaves) per mezzo di uno solo di essi (master).

Ciò permetteva allo strumentista performance che non avrebbe potuto ottenere se ogni strumento fosse stato indipendente dagli altri. Naturalmente ad un certo punto e sorto il problema della compatibilità fra le varie apparecchiature, e così si è finalmente costituito un pool di aziende costruttrici di strumenti musicali elettronici allo scopo di definire un protocollo standard di comunicazione, al fine di permettere la connessione fra strumenti di produttori diversi. Questo protocollo è stato denominato MIDI, acronimo di Musical Instruments Digital Interface (interfaccia digitale per strumenti musicali), ed è stato ben presto adattato al controllo tramite computer.

Purtroppo però la storia del MIDI è un po' più travagliata di quanto le poche righe precedenti lascino intendere: la gestione delle implementazioni MIDI è stata un po' caotica, ed alla fine lo standard si è rivelato nei fatti un po' meno standardizzato del previsto, cosa che rende talvolta problematico il colloquio tra le apparecchiature, oltre a ciò il MIDI, sorto come compromesso fra varie esigenze, è nato già obsoleto e vive senza soddisfare pienamente nessuno, tra l'altro portandosi dietro piccoli difetti di progetto quali la bassa velocità di trasmissione e la macchinosità del sistema di controllo. Ma tutto ciò a noi interessa solo di riflesso, non dovendo occuparci di questo lato della vicenda. Quello che conta è che il funzionamento è assicurato (e di buon livello) nel caso si usino dispositivi di uno stesso costruttore e non si colleghino decine di tastiere tra loro; condizioni che, crediamo, saranno rispettate in pratica, almeno dai musicisti non professionisti.

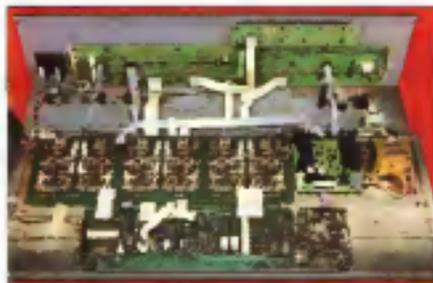
Costruttore:
SIEL S.p.A.
Zona Industriale Acquaviva Picena
63039 S. Benedetto del Tronto (AP)
Distributore per l'Italia:
Aramini Strumenti Musicali s.r.l.
Via Bruno Buozzi 1/B
40057 Cadriano Di Gran. E. (BO)
Prezzi (+IVA):
Opera 6 sintetizzatore polifonico programmabile L. 2.500.000
M.C.I. MIDI Computer Interface L. 199.000

L'interfaccia MIDI che permette di controllare un sintetizzatore con un computer.

L'interfaccia SIEL

SIEL è una giovane industria elettronica marchigiana che in nove anni di attività nel settore degli strumenti elettronici ha saputo conquistarsi una solida fama a livello mondiale. Grazie ad alcune joint-ventures con grandi produttori americani (Sequential Circuits, ad esempio, la casa dei sintetizzatori Prophet), SIEL è entrata sin dall'inizio nell'affare MIDI, facendo parte del consorzio originario di aziende che formalizzarono lo standard. Forti dell'esperienza maturata nella progettazione elettronica, e con tutto il necessario know-how del protocollo MIDI, i tecnici SIEL propongono ora un interessante prodotto destinato ad avvicinare i due mondi disgiunti dei musicisti e degli appassionati di computer: un'interfaccia MIDI, che permette ad un home computer di svolgere il ruolo di master keyboard, o controllore della catena. Il computer è stato volutamente scelto nella fascia home per non costringere il musicista all'acquisto di una macchina troppo costosa e dalle prestazioni sovrabbondanti rispetto alle reali necessità, ma in futuro verranno probabilmente sviluppate implementazioni su macchine di classe più elevata. I computer scelti sono il Commodore 64 ed il Sinclair Spertrum, in quanto già presenti nel mercato italiano con un notevole parco installato. L'interfaccia opera indifferentemente con entrambi, grazie ad un intelligente connettore bivalente, il software naturalmente deve essere quello previsto per il particolare computer connesso.

Tramite l'interfaccia MIDI il computer prende possesso del sintetizzatore cui è collegato, può modificare qualunque parametro e suonare qualunque nota, anche variando dinamicamente i parametri del suono. La trasmissione può anche avvenire in verso opposto, e quindi il computer è in grado di leggere le impostazioni dei controlli dello strumento per salvarle ad esempio su disco o presentarle sullo schermo, e può perfino riconoscere e memorizzare le

Il sintetizzatore Opera 6 della SIEL.

A sinistra la tastiera aperta. Sopra le connessioni di cui e dotata.

note suonate dal musicista per poi ripeterle in un secondo tempo, funzionando così da sequencer digitale. Naturalmente la parte del leone la fa il software che gestisce tutta la vicenda, che deve essere all'altezza della situazione. E anche da questo punto di vista ci sembra di poter dire che i signori della SIEL abbiano fatto un buon lavoro.

Il software di gestione

L'interfaccia senza l'opportuno software che ne sfrutti al meglio le potenzialità è praticamente inutile, e vero che ci si potrebbe sviluppare da soli del software su misura, ma è vero anche che questo compito risulta piuttosto complesso ed impegnativo, ed in definitiva conviene trovarsi il lavoro già fatto. Oltretutto il software fornito dalla SIEL si basa sulla profonda conoscenza del protocollo MIDI posseduta dall'azienda, e quindi c'è da ritenere che utilizzi nel modo migliore le possibilità offerte dal sistema.

Vediamo quindi in breve cosa abbiamo

Il sintetizzatore elettronico: com'è fatto, come funziona

Dunque, parliamo brevemente di cos'è un sintetizzatore elettronico. Come dice il nome, il sintetizzatore è un oggetto in grado di "sintetizzare" i suoni, ossia di crearli artificialmente per via elettronica. I suoni creati possono essere imitazioni di suoni esistenti (pianoforte, violino, ecc.) oppure suoni "nuovi" o rumori. Il suono generato risulta disponibile come voce su di una normale tastiera, e con esso si possono eseguire melodie. I primi sintetizzatori erano strettamente monofonici, ossia permettevano di suonare una nota alla volta, quelli attuali sono quasi sempre polifonici, ossia permettono di suonare anche accordi, anche se con qualche limite sul numero di due note suonate contemporaneamente (generalmente sei od otto).

Siccome il "timbro" di un suono dipende dalla sua forma d'onda, la sintesi operata da un sintetizzatore consiste nel generare un segnale elettrico avente un andamento uguale (in teoria) alla forma d'onda del suono da riprodurre; siccome questo non è possibile, si cerca comunque di far sì che l'onda generata sia il più possibile simile a quella originale. Per fare ciò ci si basa su di un teorema dovuto a Fourier che afferma che ogni forma d'onda complessa può essere sempre costruita come somma di infinite forme d'onda sinusoidali, aventi frequenze multiple di una data (detta fondamentale) ed ampiezze opportune. Sfruttando questo risultato si può quindi giungere in linea di principio ad approssimare un suono quanto si voglia, usando forme d'onda semplici come le sinusoidi. Nella pratica ciò si può realizzare secondo due diverse strategie, dette sintesi additiva (o per formanti) e sintesi sottrattiva. La prima applica letteralmente il teorema di Fourier, ricostruendo il suono come somma di tante sinusoidi elementari. Operativamente si deve poter disporre di un banco di oscillatori sinusoidali in frequenza ed ampiezza, le cui uscite vanno miscelate insieme. Naturalmente questa è una via assai complessa e dispendiosa, e pertanto poco utilizzabile nella pratica.

La seconda si basa su di un altro teorema di teoria dei segnali che afferma che un'onda quadra si ottiene come somma di infinite sinusoidi aventi le ampiezze in un determinato rapporto (noto, ma che non ci interessa). Imprecisamente ciò equivale a dire che un'onda quadra possiede "tutte" le sinusoidi. Ed allora si può sfruttare questo fatto per generare qualunque altro suono a partire da un'onda quadra, semplicemente "cancellando" da essa tutte le componenti che non interessano e lasciando solo quelle necessarie. Ciò in pratica si realizza mediante un solo oscillatore ad onda quadra la cui uscita viene inviata ad opportuni filtri (passa alto, passa banda, passa basso) che provvedono ad attenuare le componenti indesiderate. Se volete, potete paragonare questi due metodi ai due modi di realizzare una scultura: se lo scultore lavora con la creta opera "additivamente", creando una forma che prima non c'era, se invece scolpisce il marmo opera "sottrattivamente", man mano eliminando dal blocco informe tutte le parti che non appartengono al soggetto.

La sintesi formante permette una mag-

Lo schema a blocchi del sintetizzatore Opera 6. In esso non si può inviare l'LFO al VCA mentre in altre macchine ciò è possibile.

a disposizione. Il primo prodotto in ordine di tempo ad apparire sul mercato è stato il Multitrack Composer, un editor musicale per C64 piuttosto semplice, ma comunque efficace. I moduli più recenti sono invece assai più sofisticati e raggiungono un elevato livello di professionalità. Abbiamo visto ad esempio un ottimo Live Sequencer a sedici tracce, che permette di simulare un registratore da studio e costruire da soli un brano registrando le varie tracce in successione, la memorizzazione avviene in tempo reale ed in polifonia, e si hanno piene possibilità di controllo sulle sequenze registrate. Un altro prodotto assai interessante è il Data Base Synthesizer, un archivio di timbri. Esso permette di memorizzare su disco il contenuto delle memorie di un sintetizzatore programmabile, realizzando così un archivio costituito dai parametri di controllo che formano un dato timbro. Gli strumenti programmabili consentono soltamente il dump su cassetta del contenuto delle memorie, ma il metodo non è standardizzato fra i vari prodotti, in questo modo, invece, grazie alla connessione MIDI si possono salvare i timbri di strumenti di case diverse, con ovvio guadagno in sintesi e semplicità operativa. Il programma permette inoltre di archiviare i file per classi, di gestire sequenze di caricamento particolari per configurare interi banchi di strumenti e così via. Abbiamo poi un echo delay digitale ed un Keyboard Multitracking, disponibili solo per C64, molto utili nelle performance dal vivo: il primo genera un effetto eco/riverbero programmabile via MIDI, il secondo consente sofisticate possibilità di split della tastiera assegnando ad ogni sezione i timbri impostati su di un expander. E per finire abbiamo diversi Sound Editors adatti ai vari modelli di tastiere e/o expander prodotti dalla SIEL, e disponibili anche per Spectrum. Con essi si possono visualizzare, variare e memorizzare i valori di tutti i parametri di controllo

La meccanica della tastiera che grazie al doppio contatto permette all'elettronica di "capire" la forza con cui viene premuto il tasto. A destra un'immagine del programma che simula i controlli della tastiera

giore precisione di ricostruzione a scapito di una notevole complessità di gestione, quella sottrattiva è invece più economica anche se meno precisa. Generalmente, tranne rari casi appartenenti al mondo della ricerca, si sceglie la seconda, che consente ugualmente buoni risultati con un hardware piuttosto semplice e poco ingombrante. Tutti i sintetizzatori commerciali, in particolare, funzionano così.

Come esempio dell'altra tecnica citiamo invece il caso dell'IRCAM di Parigi, dove è stato costruito appositamente un megasintetizzatore nel quale ogni suono viene ricostruito mediante somma dei segnali prodotti da un banco di 255 oscillatori sinusoidali indipendenti, il tutto (ovviamente) sotto il controllo di un computer.

Un'altra questione da affrontare nella sintesi dei suoni è quella dell'inviluppo di ampiezza. Un colpo di tamburo, ad esempio, ha un andamento nel tempo ben diverso dal suono di un flauto. Il primo è un suono impulsivo, il secondo non lo è, la differenza fra i due strumenti è notevole, anche se timbricamente i suoni emessi si assomigliano essendo entrambi pressoché sinusoidali. È necessario quindi anche modulare l'ampiezza del suono nel tempo per conferirgli una maggiore realtà. Ciò viene fatto mediante un generatore di inviluppo sincronizzato alla tastiera, la cui uscita va a pilotare un apposito amplificatore cui spetta il compito di modulare il segnale proveniente dall'oscillatore. L'inviluppo è una particolare forma d'onda che ripete l'andamento dell'intensità del suono nel tempo, ed i cui parametri (detti Attack, Decay, Sustain e Release, ossia tempo di attacco, tempo di primo decadimento, livello di mantenimento e tempo di rilascio), sono regolabili a piacere. Il generatore di inviluppo è un particolare circuito che, ad ogni pressione di tasto, emette una forma d'onda con questi parametri.

Riassumendo, un sintetizzatore è formato da uno o più VCO (Voltage Controlled Oscillator, oscillatore controllato in tensione) che emettono il segnale (generalmente a scelta fra onda quadra, triangolare e dente di sega), uno o più VCF (Voltage Controlled Filter, filtro controllato in tensione) che modificando il contenuto armonico del segnale, ed uno o più VCA (Voltage Controlled Amplifier, amplificatore controllato in tensione) che modulano in ampiezza il segnale. Come dicono gli stessi nomi, ogni modulo è controllabile in tensione, il che significa che si può variare ad esempio la frequenza di taglio del VCF o il livello di uscita del VCA inviando loro una opportuna tensione di controllo. Queste tensioni possono essere fisse, regolate da potenziometri posti sul pannello, o variabili, generate da appositi circuiti. Il più semplice è LFO (Low Frequency Oscillator, oscillatore a bassa frequenza), che a seconda della destinazione può creare effetti di tremolo, vibrato o wah-wah. Un altro è proprio il generatore di inviluppo o ADSR (Attack Decay Sustain Release), il cui scopo principale è pilotare il VCA ma che può anche controllare il VCF. Nella filosofia originaria di Robert Albert Moog, il creatore dei primi sintetizzatori, ogni modulo doveva poter pilotare qualunque altro, pertanto le connessioni fra le varie sezioni venivano effettuate per mezzo di jack (e i sintetizzatori assomigliavano più ad un centralino telefonico che ad uno strumento musicale), e le tensioni di pilotaggio erano tutte standardizzate in un range 0-5 volt. Gli attuali sintetizzatori differiscono un po' dai loro predecessori, ma i concetti di base sono rimasti gli stessi.

Il sintetizzatore modello Opera 6 della SIEL, consegnatoci per la prova assieme all'interfaccia MIDI ed al suo Expander, non sfugge alla regola. In figura ne potete vedere un sommario schema a blocchi. L'uscita dell'oscillatore (che in realtà sono più d'uno, per ottenere la polifonicità ed un timbro più "pieno") viene inviata ad un VCF (un passabasso ad elevata pendenza), e da qui al VCA. L'ADSR può pilotare il VCA, il VCF o entrambi, e sono disponibili due LFO che modulano i VCO e/o il VCF. Nelle foto vedete anche l'interno dello strumento. In particolare si può notare come i moduli "tutti analogici" siano pochi, essendo tutti i circuiti controllati digitalmente. Naturalmente la "marcia in più" dell'Opera 6 e degli strumenti analoghi è costituita dal controllo MIDI. Esso in pratica si sostituisce ai potenziometri posti sul frontale, permettendo di pilotare ogni modulo dall'esterno in modo digitale, sia per mezzo dei controlli di un altro sintetizzatore sia da computer. In questo modo i vari parametri possono essere variati dinamicamente durante l'esecuzione, ottenendo effetti molto sofisticati. Tramite il bus MIDI ed un opportuno software di gestione, quindi, un computer può controllare un sintetizzatore predisposto ed intervenire sui vari parametri di controllo in modi e tempi che uno strumentista umano, dotato di solo dieci dita, due mani e due piedi, difficilmente potrebbe imitare.

Una visita alla SIEL

Allo scopo di renderci conto più da vicino della realtà SIEL abbiamo un bel giorno deciso di fare un salto in quel di San Benedetto del Tronto. Giunti a destinazione ci ha accolti nella moderna sede dell'azienda il simpatico e barbuto Mario Picchio, EDP manager della SIEL ed ora responsabile della SIEL Software Division, che si occupa appunto di produrre e commercializzare i supporti al MIDI, sotto forma di prodotti hardware e software. Purtroppo lo spazio limitato non ci consente di occuparci a fondo di tutto ciò che abbiamo visto e ci siamo detti in una intera giornata, possiamo però dire di aver riportato l'impressione di trovarci di fronte a gente ben preparata e che sa il fatto suo, ad un'azienda giovane e dinamica che ha imboccato una strada vincente.

Le novità fervono, in casa SIEL, quando la rivista sarà in edicola sarà da poco cominciata la distribuzione del nuovo sintetizzatore DK 80, e dell'unità CMK 49. Accenniamo brevemente a quest'ultima perche pensiamo possa interessare tutti i Commodoriani. Si tratta di un sistema composto da una tastiera musicale a 49 tasti e da un apposito software, che gestisce il SID del 64 facendolo diventare un sintetizzatore. Sono in particolare disponibili tutti i controlli presenti sui "veri" strumenti quali l'Opera 6, compresa la possibilità di memorizzare i timbri creati come preset. La tastiera si applica all'uscita cartridge (che duplica), e consente contemporaneamente la connessione MIDI. Il tutto ad un prezzo ufficioso che non dovrebbe superare le trecentomila lire.

Curiosando fra le linee di produzione e gli uffici di ricerca e sviluppo abbiamo poi scoperto un oggetto veramente interessante, benché ancora allo stato di prototipo. Sarà una scatola...suonante, controllata da MIDI, e con delle funzioni sorprendenti. Non possiamo ancora dire di più, ma da ciò che abbiamo visto pensiamo che quando questo oggetto uscirà sul mercato farà un grosso colpo nel mondo della microinformatica musicale.

Ripartiamo a tarda sera, con diversi depliant e molte notizie, soddisfatti della giornata. Pensiamo che della SIEL e dei suoi prodotti risentirete presto parlare su queste pagine, e non solo per la prova del CMK 49.

A sinistra, spionaggio industriale negli uffici R&D della SIEL: un prototipo funzionante della scatola misteriosa descritta nel testo. A destra, un bel rack con strumenti e computer. Sotto, due particolari delle linee di produzione degli strumenti: il test dell'elettronica, che avviene mediante un banco apposito e la prova di "cottura" in cui ogni tastiera rimane alimentata per quarantotto ore.

Tre particolari delle varie sezioni di controllo della tastiera. Tutti i valori passono essere settati via MIDI. A sinistra i tasti per l'impostazione delle 100 memorie Preset.

degli strumenti, consentendo all'utilizzatore di creare i suoi propri timbri che verranno immagazzinati nelle memorie dello strumento per essere poi usati nell'esecuzione. La versione per la nuova tastiera DK 80 sfrutta pienamente le capacità grafiche del computer su cui gira, mostrando ad esempio in alta risoluzione la forma d'onda risultante dalle varie impostazioni effettuate, l'inviluppo d'ampiezza selezionando e così via, il tutto praticamente in tempo reale. Quella per l'expander dell'Opera 6 sfrutta ugualmente la grafica, ma per simulare il pannello di controllo della tastiera, mostrando posizione e valore dei potenziometri. Alcuni di questi programmi ci sono stati mostrati durante la nostra visita alla SIEL (vedi riquadro), come impressione generale possiamo dire che ci è parso di rilevare una grande cura nella definizione dei prodotti, essenzialmente dovuta alla sintesi di esperienze sia tecniche che musicali ottenute nel tempo dalla SIEL.

Considerazioni finali

Al termine di questa prova inusuale buttiamo giù qualche conclusione. Ci sembra innanzitutto che un prodotto come questo sia degno del massimo interesse per le potenzialità di "culturizzazione" che porta con se. Può essere un ottimo mezzo per avvicinare i musicisti al computer e gli appassionati di computer alla musica, in modo semplice e non traumatico per entrambi. La scelta di sviluppare prodotti per computer di basso costo e larga diffusione ci sembra a riguardo piuttosto sintomatica. C'è da dire che il costo stesso dei prodotti SIEL è mantenuto quasi a livelli politici, proprio per facilitare al massimo la penetrazione in profondità di questi oggetti e delle problematiche che li accompagnano. L'interfaccia costa meno di duecentocinquantamila lire, i vari software dalle cinquanta alle centocinquanta. La tastiera Opera 6 costa parecchio di più, circa due milioni e mezzo, ma SIEL produce altri sintetizzatori programmabili di taglio più consumer e di costo più ridotto, quali il nuovo DK 80 da pochi giorni in distribuzione, che costa circa un milione e ottocentomila lire, o il simpaticissimo MK 900, portatile autoamplificato con batteria programmabile incorporata, che costa poco più di un milione. Come si vede non sono prezzi inavvicinabili. Inoltre per chi volesse provare l'ebbrezza della musica elettronica senza comprare un sintetizzatore, è disponibile il nuovo CMK 49,un add-on per C64 che ne sfrutta il SID interno come sintetizzatore, grazie ad un apposito software e ad una tastiera musicale vera che si collega al computer, ad un prezzo che viaggia attorno alle trecentomila lire.

Insomma, crediamo che la SIEL abbia intrapreso la strada giusta, anche alla luce di alcune novità intraviste in fabbrica e di cui riparleremo al momento opportuno. Con il complesso di prodotti hard/soft SIEL ogni possessore di C64 o Spectrum che sia anche un tantino musicofilo può permettersi di mettere su uno studiolo di computer music senza per questo spendere cifre folli o installare attrezzature da IRCAM. E, viceversa, un musicista dilettante (o professionista) può avvalersi del computer senza dover acquistare un VAX per i suoi esperimenti. Crediamo che questa operazione permetta da un lato ai musicisti di aggiornarsi professionalmente giungendo a nuovi standard di lavoro, e dall'altro, che ci riguarda forse più da vicino, metta a disposizione dei possessori di computer un modo intelligente di sfruttare la propria macchina, quando si saranno stancati magari di farci i videogiochi. MC

L'interfaccia aperta e le sue connessioni

di Corrado Giustozzi

Siamo quindi giunti al nostro terzo ed ultimo appuntamento con Life, l'avvincente gioco di simulazione al calcolatore inventato da J. H Conway. Nelle puntate precedenti abbiamo visto in dettaglio scopi e regole del gioco e modi di implementazione su un personal, passando poi in rassegna alcune interessanti strutture dal comportamento piuttosto particolare. Questo mese termineremo il discorso facendo conoscenza con nuovi oggetti altrettanto peculiari, ed infine accenneremo a qualche estensione e/o variazione alle regole della simulazione che conducano a classi di giochi simili a Life nel concetto, ma diversi nel comportamento. La volta scorsa, se vi ricordate, parlammo di quel gruppo di ricercatori del M.I.T. autore di diverse scoperte sullo strano mondo di Life. Fra le creature da essi generate avevamo visto in particolare il mostruoso cannone ad alianti, che dà vita ad una infinità di individui che si spostano sul piano alla velocità di una cella ogni quattro generazioni. Bene, non contenti di ciò, i nostri scienziati misero a punto successivamente una struttura in grado di catturare e distruggere gli alianti emessi dal cannone! La vedete in figura 1, non lasciatevi ingannare dal suo aspetto semplice e lineare, ci troviamo di fronte ad un terribile mangiatore di alianti! (I suoi scopritori lo battezzarono pentadecatlon, in quanto si tratta di un oscillatore con periodo quindici). A seconda delle posizioni relative del pentadecatlon e dell'aliante, quest'ultimo può essere assorbito o, al contrario, venire riflesso e quindi "rimbalzare" all'indietro. Queste due possibilità possono essere entrambe sfruttate per ottenere costruzioni sorprendenti. Nel primo caso si può fare in modo che un pentadecatlon opportunamente collocato intercetti l'intero flusso di alianti emesso dal cannone, assorbendolo completamente, altrimenti si può creare una struttura surreale formata da due pentadecatlon che giocano a ping pong, rilanciandosi un aliante avanti e indietro.

Per la cronaca, esistono diversi tipi di alianti a cinque cellule, oltre a quello visto la volta scorsa potete osservarne un altro in figura 2. Il suo comportamento è esattamente analogo a quello dell'altro esemplare, ma questo tipo si differenzia dal precedente in quanto è alla sua razza che appartengono gli individui emessi dal cannone e mangiabili dal pentadecatlon.

Se a questo punto credete che questo sia il massimo e non sia possibile ottenere di più avete ben poca fiducia nei cervelloni del M.I.T.! Fra i virtuosismi di questo gruppo di Life-maniaci si annoverano oggetti che definire incredibili è poco. Per esempio, avreste mai pensato che fosse possibile collocare due o più cannoni ad alianti in modo che i vari individui in movimento giungessero ad una collisione tale che dall'interazione sorgesse una struttura a sua volta in grado di costruire e lanciare alianti di tipo diverso? Bene, anche ciò è stato fatto. Di questa struttura fantastica vi mostriamo solo l'ultimo stadio, ossia gli alianti prodotti, sono del tipo ad otto cellule riprodotto in figura 3, ed il gruppo del M.I.T. li ha denominati astronavi leggere per distinguerlo dal gruppo degli alianti a cinque cellule.

Sempre in tema di alianti se ne mettete due come in figura 4, il risultato sarà la distruzione completa di entrambi: alla quarta mossa non rimane nessuna cellula viva. È possibile ottenere una cosa analoga per le astronavi? Sì, basta disporle come in figura 5, la morte totale avviene in questo caso alla settima mossa.

Come ormai già sapete, esistono in Life delle particolari configurazioni che non si evolvono affatto, ossia non mutano da una generazione

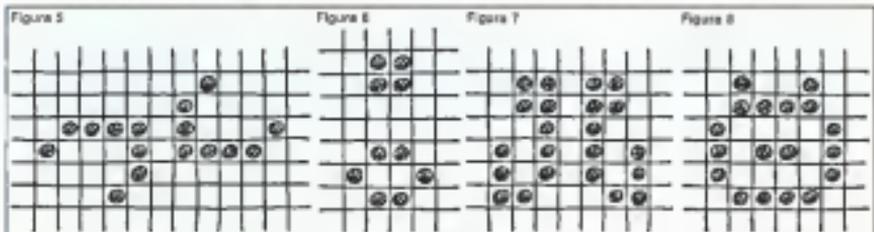
Figura 1

Figura 2

Figura 3

Figura 4

Figura 1 - Un terribile mangia-alianti in grado di divorare un flusso di alianti come quelli di figura 2. Figura 2 - Un aliante a cinque cellule parente di quello vista la volta scorsa. Sono di questo tipo gli alianti emessi dal cannone lancia-alianti. Figura 3 - Un astronave leggera ossia un aliante ad otto cellule. Figura 4 - Un cruento scontro di alianti, alla quarta generazione non vi sarà più nessuna cellula viva.

all'altra, esse sono dette stabili, e ne vedete due esempi in figura 6 il quadrato ed il favo. Una zona di mondo interamente costituita da ripetizioni di queste due strutture, purché le distanze reciproche non siano inferiori ad un minimo di "non interazione", costituisce un campo anch'esso stabile. Provate allora a costruire un simile campo e, una volta accertatane la stabilità, ad inserire uno ed un solo nuovo individuo in un punto qualunque della struttura. Cosa succede? Il nuovo arrivato crea senz'altro scompiglio nella configurazione, che comincia a perdere il suo equilibrio, poi, a seconda della geometria iniziale e della posizione dell'intruso, o tutto torna a posto e l'elemento perturbatore viene eliminato, oppure tutto l'ordine viene distrutto e la struttura originale si disfa nel caos. Ecco scoperto un interessante modello che rappresenta, ad esempio, l'aggressione di una coltura cellulare da parte di un virus: può darsi che la colonia attaccata sia in grado di sconfiggere l'intruso, ma più di frequente è quest'ultimo ad avere la meglio distruggendo l'intera colonia.

Abbiamo parlato più volte di oscillatori, un oscillatore nel mondo di Life altro non è che una configurazione avente la proprietà di tornare identica a se stessa dopo un certo numero di generazioni. In altre parole un oscillatore è una struttura la cui evoluzione consiste solo nell'assumere ciclicamente una medesima successione di forme e posizioni nel piano. Il suo periodo è il numero di generazioni dopo le quali ritorna alla configurazione di partenza. Il più semplice oscillatore l'abbiamo incontrato la volta scorsa: è formato da tre individui in linea retta ed ha periodo due (ossia è un flip-flop). Il gruppo del M.I.T., fra le altre cose, ha costruito anche tutta una serie di oscillatori dai comportamenti complessi e strabilianti. In figura 7 ve ne presentiamo uno assai simpatico, detto "l'acrobata", il suo numero consiste nel fare le capriole, o meglio volteggiare attorno a se stesso, una volta ogni sette. E terminiamo con un piccolo capolavoro, detto Gatto del Cheshire. Chi ha letto Alice nel Paese delle Meraviglie conosce senz'altro il famoso Gatto del Cheshire, che svanisce lasciando visibile solo il sogghigno. Ebbene, in figura 8 vedete appunto il muso del Gatto (prima che svanisca, ovviamente). Provate a ricostruirne l'evoluzione. Avrete la sorpresa di vedere ben presto il muso dissolversi in configurazioni piacevolmente simmetriche che, alla sesta generazione, danno inaspettatamente luogo all'apparizione del sorriso, dopodiché anche questo sparisce ed al suo posto rimane, stabile, l'impronta della zampa del Gatto!

E con questa meraviglia finale giungiamo all'ultima parte del nostro viaggio nel mondo di Life. Resta da vedere come si possono estendere o modificare le regole del gioco per ottenere diverse varianti nel comportamento degli organismi in esame. Cominciamo col notare che esistono diverse categorie di regole applicabili ad una situazione "tipo Life". E per capire le possibili varianti approfondiamo un po' la questione. La regola di base è che lo stato futuro di una cellula deve essere stabilito in base a quello dei suoi vicini (eventualmente inclusa come caso particolare la cellula stessa). Già però la nozione di "stato" della cellula è suscettibile di ampliamenti. In Life esistono solo due stati: le cellule possono essere solo vive o morte, e le regole stabiliscono appunto quando ogni cellula debba cambiare di stato nascendo o morendo. Nulla impedisce però che le cellule possano assumere più di due stati, tre, ad esempio, che potremmo chiamare bianco, nero e grigio, o anche di più. In questo modo le regole possono consentire o impedire le possibili transizioni di stato in base ad eventi più complessi (anche i vicini hanno più stati, ovviamente, e questo complica le cose).

E già che siamo in tema, notiamo che anche per il modo di scegliere i vicini si possono seguire due regole: considerare tutte e otto le cellule adiacenti a quella in esame (due in verticale, due in orizzontale e quattro sugli spigoli) oppure restringere l'attenzione alle sole quattro che confinano con essa mediante un lato (nord, sud, est e ovest). Nel primo caso, che è quello di Life, si dice che di ogni cellula si considera l'intorno di Moore, nel secondo caso l'intorno di Von Neumann, in onore ai due studiosi che hanno gettato le basi rigorose della teoria degli automi (e Life rappresenta proprio un esempio di automa cellulare, come dicemmo nell'introduzione a questa serie di articoli). Già vediamo che le cose sono diventate parecchio complesse. Ma ancora non è finito: rimane da scegliere il tipo di regole di transizione. Nell'ambito di ognuno dei due intorni si possono applicare diversi tipi di regole. Le due classi principali si chiamano totalistiche (o di conto) e geometriche. Esse stabiliscono lo stato della cella in base a quello dei vicini in modo rispettivamente "quantitativo" e "qualitativo". In base a queste considerazioni si possono costruire innumerevoli esempi di giochi tipo Life. Nella prossima puntata, quindi, ne vedremo qualcuno, entrando più in dettaglio nei criteri di scelta delle regole. MC

Figura 5 | Figura 6 | Figura 7 | Figura 8

Figura 5 - Stesso scenario di figura 4, in cui però sono coinvolte due astronavi leggere. Figura 6 - Due figure stabili, ossia che non mutano da una generazione all'altra: sopra il quadrato, sotto il favo. Figura 7 - L'acrobata, che ogni sette generazioni si rovescia su se stesso. Figura 8 - Un capolavoro: il Gatto del Cheshire. Come quello di Alice svanisce lasciando solo il sogghigno!

Se è vero che esistono giochi che piacciono "al volo", ebbene questo è uno di quelli Come molti sapranno, esso prende spunto dall'omonimo e divertentissimo film, in programmazione in molti cinema, ed i vari copyright vengono rimarcati in una apposita schermata, la prima, che rimane presente per quasi tutto il tempo di caricamento del gioco dal disco

La cosa che attrae di più, e vi assicuriamo in maniera decisiva, è la musica, la stessa del film, scritta (e cantata, ma non in questa sede) da Ray Parker Jr , che ci allieta durante tutto lo svolgimento del gioco Su di essa è possibile canticchiare le varie strofe che ci vengono mostrate in successione in una delle schermate.

Ma vediamo brevemente la storia dei Ghostbusters Due studiosi di fenomeni paranormali e diavolerie affini vengono buttati fuori dall'Università per scarso rendimento e decidono allora, insieme ad un altro amico, di investire tutti i loro soldi mettendo su una ditta di acchiappafantasmi Il successo è quasi immediato perchè, guarda caso, New York viene infestata dagli spettri che i nostri eroi, grazie ai loro diabolici marchingegni, riescono, dopo varie vicende, prima a catturare ed infine a scacciare dalla terra

Passiamo ora a descrivere sommariamente lo svolgimento del gioco Esso si articola su un gran numero di schermate, alla prima abbiamo già accennato, La seconda presenta lo stemma dei Ghostbusters, un fantasmino (simpaticissimo!) che sbuca da una "O" sbarrata, sotto la quale scorrono le parole della canzone di Parker, introdotta dalla frase — "Ghostbusters ha ha, ha " — sintetizzata con il computer Si e quindi invitati a

premere f1 o f3 per iniziare il gioco Premuto uno dei tasti indicati appare una prima schermata che ci accoglie nel club degli acchiappafantasmi Ci viene chiesto se abbiamo un conto a nostro nome (n.b , se durante le partite precedenti abbiamo guadagnato un certo gruzzolo catturando un bel po' di fantasmi e distruggendo il Marshmallow, abbiamo la possibilita di aprire un conto a nostro nome, presso la banca della citta, che viene gestito automaticamente su disco, quindi nulla di ciò che si e conquistato va sprecato) In caso contrario ci vengono elargiti 10000 dollari per l'acquisto dell'equipaggiamento utile per la cattura dei fantasmi

Anche la scelta di tale equipaggiamento si articola su più schermate, selezionabili da tastiera, tre delle quali sono riservate alla scelta di una adeguata autovettura ed altre tre ci mostrano vari attrezzi utili per la cattura degli spettri tra cui compaiono sensori per fantasmi, esche, intensificatori di immagini pompe aspirafantasmi, laser, trappole (queste ultime da acquistare obbligatoriamente) ecc

Naturalmente con 10000 dollari a nostra disposizione non potremo compare tutto subito, ma potremo farlo in un secondo tempo quando cioe la nostra abilita di cacciatori ci permetterà di catturare un buon numero di fantasmi, cosa che produrrà la crescita del capitale e ci permetterà di scegliere un'attrezzatura più sofisticata

Fatte le nostre scelte, premendo il tasto "E" passeremo nel vivo del gioco Ci verrà mostrata una piantina della città e dai quattro lati del teleschermo vedremo spuntare quattro fantasmi che dovremo bloccare intercettandoli con un cursore (lo stemmino dei Ghostbusters) che muoveremo servendoci del joystick I fantasmi si muoveranno in un primo momento lentamente, ma la loro velocità aumentera durante il gioco, insieme all'energia "spettrale" della città, indicata da una apposita linea di stato sul teleschermo Se non riusciamo ad intercettare in

tempo gli spettri, essi si riuniranno in un edificio, contrassegnato dalla scritta Zuul, e produrranno un aumento dell'energia cui accennavamo oppure insieme, quando l'energia supererà un certo limite, formeranno un grande pupazzo, il Marshmallow, dalla grande potenza devastatrice il quale ci porterà via una buona fetta di capitale (potremmo fermare i fantasmi che vanno a formare il pupazzone catturandoli con un'esca, premendo il tasto "B", ma della strategia del gioco non ci occuperemo).

A questo punto dovremo portare il nostro cursore in prossimità di uno degli edifici che lampeggiano sulla pianta e, indicandolo con la leva del joystick, premeremo il fuoco. Ci ritroveremo a percorrere, con la nostra autovettura, una strada lungo la quale dovremo cercare di acchiappare i quattro fantasmi che incontreremo durante il cammino (ogni fantasma perduto produrrà un aumento dell'energia a nostro sfavore) dopo di che l'autovettura si fermerà e comparirà una schermata che ci presenta la facciata dell'edificio selezionato. Su di esso, se siamo fortunati, vedremo aleggiare un fantasma che dovremo catturare con i raggi antifantasma (che non vanno mai incrociati) e con le trappole che posizioneremo abilmente facendole scattare al momento opportuno premendo il fuoco.

La parte più avvincente del gioco comincerà quando, ad un certo punto (energia dei fantasmi 10000 e somma accumulata superiore ai 10000 dollari), una chiave ed una serratura, che scorrono sullo schermo, si incontreranno e noi dovremo entrare nello Zuul, evitando il grande Marshmallow che comparirà sullo schermo, e da lì sconfiggere le forze del male. Dimenticavamo di dirvi che ogni qual volta terminerete le trappole o altro dovrete andare a rifornirvi portandovi in prossimità dell'edificio contrassegnato delle lettere "GHQ", il quartier generale dei Ghostbusters, e premere il joystick.

Non ci dilunghiamo oltre sullo svolgimento del gioco, sia perché lo spazio non ce lo concede, sia perché vogliamo lasciarvi con un pizzico di curiosità. Crediamo comunque di aver reso l'idea di tutto l'insieme di elementi che, uniti ad una grafica di ottima qualità, rende il gioco piacevolissimo. *T.P.*

Produttore:
Activision International
Mountain View CA 94039 - USA

Distributore per l'Italia:
Miwa Trading s.r.l.
Centro Direzionale Milanofiori
Strada 7 - Palazzo T1
20089 Rozzano (MI)

Artworks
STRIP POKER
Commodore 64

I giocatori di poker non rimarranno delusi da questo gioco il quale ci mette in competizione con la macchina che si comporta abbastanza intelligentemente (si fa per dire): pensate che bluffa anche!

La nostra avversaria è una provocante fanciulla, Suzi o Melissa (a scelta del giocatore), che potremo selezionare servendoci del joystick.

Il gioco si svolge seguendo le regole di una normale partita di poker a 52 carte: la novità consiste nel fatto che la nostra avversaria, da buona giocatrice, non si

tira mai indietro se è convinta di avere una mano vincente e, in mancanza di liquidi, vende i propri indumenti per 100 dollari al pezzo pretendendo lo stesso anche da noi, naturalmente se stiamo perdendo.

Veniamo allo svolgimento del gioco. I due avversari, che avranno una posta iniziale di 100 dollari a testa, distribuiranno alternativamente le carte — che vedremo comparire in basso sul teleschermo — e, chi le riceve per primo, inizierà il gioco. Egli, in base alle carte ricevute, ha tre possibilità: puntare (Bet), passare parola (Stay) o passare definitivamente (Drop). Le varie scelte vengono selezionate mediante la cloche del joystick e rese attive dal tasto del fuoco. Se passiamo parola, ci sono due possibilità: l'avversaria può o passare anch'essa oppure aprire il gioco con una sua puntata. La stessa cosa si verifica se siamo noi ad aprire. Se entrambi i giocatori decidono di giocare, si passa alla eventuale sostituzione delle carte posizionandosi su quelle da cambiare, servendosi del joystick, e premendo il pulsante del fuoco. A questo punto per ricevere le nuove carte dovremo, manovrando la leva del joystick, far apparire la scritta "Finished" e premere ancora il fuoco.

Si passa allora alla seconda fase. In base alle carte ricevute, chi ha aperto, comincerà a "parlare" e potrà o passare, o far "parole" o puntare. A questo punto l'avversario o andrà a vedere la mano (Call) o rilancerà (Raise). Vengono quindi mostrate le carte di Suzi o di Melissa, a

seconda della giocatrice selezionata, e chi vince incasserà l'ammontare delle puntate. Se durante il gioco un giocatore si trova in difficoltà finanziarie, gli verrà dato del denaro in cambio dei propri indumenti (in verità i nostri vengono estorti quasi con la forza perché il computer non ci permette di ritirarci). Per ogni 100 dollari persi, la nostra avversaria si toglierà di dosso l'indumento posto in vendita e verrà caricata da disco una schermata successiva, che la vede sempre più nuda.

La grafica è notevole, non mancano le musichette da strip tease, i commenti salaci del computer ed i messaggi ironici verso il perdente che aggiungono, al danno, la beffa!

T.P.

Produttore:
Artworks

Che la forza sia con voi: ecco finalmente il gioco che permetterà a tutti gli appassionati della saga di Guerre Stellari di trasformarsi in altrettanti Luke Skywalker.

Anche se il disegno sulla confezione non lo lascia affatto supporre (probabilmente per problemi di copyright), 3D Starstrike è chiaramente basato sulla sequenza finale del famoso film di Lucas, quella in cui i caccia della flotta ribelle riescono a distruggere la Morte Nera.

Il gioco consiste dunque nel pilotare un caccia spaziale all'attacco della base nemica, per tentare di farla esplodere colpendo i suoi reattori nucleari. Lo svolgimento segue molto da vicino la trama del film; per prima cosa bisogna raggiungere la Morte Nera, cercando di abbattere il maggior numero di caccia nemici. Arrivati sulla gigantesca astronave si deve iniziare la fase di volo radente, fino ad arrivare all'imbocco del condotto di raffreddamento della camera dei reattori. Per questa strada si deve infine giungere ai reattori e tentare di colpirli; se la missione ha esito positivo ci si allontana automaticamente dalla Morte Nera per assistere alla sua esplosione, altrimenti si ritorna nel condotto per un nuovo tentativo.

Il tutto viene vissuto con un realismo impressionante, tenuto conto delle limitazioni di un home computer qual è lo Spectrum.

La grafica, di tipo vettoriale, è veramente entusiasmante e le foto non rendono pieno merito alla sua qualità: nono-

stante la discreta complessità delle scene la velocità di aggiornamento dello schermo è elevatissima e l'azione si volge effettivamente in tempo reale.

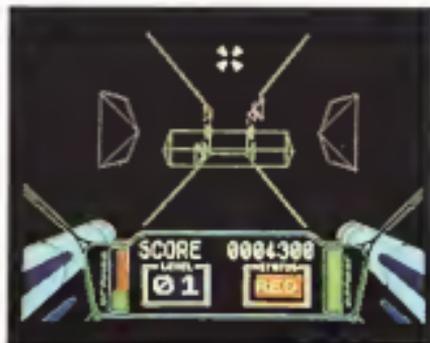
La dotazione della propria astronave consiste in due cannoncini laser e in uno schermo di protezione; i primi due sono praticamente infiniti, poiché si ricaricano automaticamente quando il pulsante di fuoco non è premuto, mentre il secondo si indebolisce ad ogni colpo ricevuto. Quando lo schermo cede definitivamente la partita termina e si deve ricominciare da capo.

Sullo schermo compare un mirino che può essere controllato sia da tastiera che da joystick: mirare con precisione è piuttosto difficile e abbiamo trovato che è meglio sparare a raffica piuttosto che a colpo singolo, facendo attenzione a rilasciare il pulsante appena è possibile per permettere ai laser di ricaricarsi.

La sequenza più affascinante è senza dubbio quella del viaggio attraverso il

condotto di scarico; qui, oltre ad evitare i colpi sparati dalle postazioni di difesa della Morte Nera bisogna anche cercare di non andare ad urtare contro gli ostacoli fissi che ostruiscono il passaggio.

I comandi sono essenziali e comprendono soltanto quelli già citati di spostamento del mirino e di fuoco; a proposito di mirino dobbiamo far rilevare che nella prima sequenza, con lo schermo affollato di caccia nemici e di colpi in arrivo, è abbastanza facile perdere di vista la sua posizione. *M B*

Produttore:
Realtime Software
Pronect House
32 Sovereign Street
Leeds (GB)

Gli Orola sono delle strane creature saltellanti di colore giallo, che a causa di strane vicissitudini si trovano sparpagliate per la galassia

Il compito del giocatore, che impersona un "Guerriero delle Biciclette Stellari" (potete anche mettervi a ridere, ma la traduzione di Starbike Warriors è proprio questa) è quello di visitare uno ad uno i pianeti su cui si presume si trovino degli Orola, per portarli in salvo sul loro pianeta natale

Naturalmente questi pianeti sono infestati dalla solita miriade di creature pericolosissime, il cui contatto va evitato ad ogni costo

La Starbike pilotata ha, come arma di difesa, un cannoncino permanentemente attivato, il giocatore può controllare invece la direzione del movimento e la spinta verticale dei razzi, che permette di volare più o meno alti rispetto allo superficie

Come gioco Starbike è nella media, ad una certa mancanza di originalità contrappone una grafica curata ed una buona velocità d'azione Notevoli le possibilità di interfacciamento con i più popolari add-on, il menu principale prevede infatti l'impiego sia del generatore di effetti sonori della Fuller che del sintetizzatore vocale della Currah

M B

Produttore:
The Edge
31 Maiden Lane
Covent Garden
London (GB)

Elite
THE FALL GUY
Spectrum

Dopo un brillante esordio con Kokotoni Wilf, già recensito sulle pagine di questa rubrica, la Elite ritenta la scalata alle classifiche con questo The fall guy, ispirato all'omonima serie televisiva con Lee Majors

Il titolo letteralmente significa "il cascatore", e nel gioco si deve appunto impersonare uno stuntman impegnato a guadagnarsi il pane saltando su una serie di oggetti in movimento treni, carri armati, casse e così via

Il film completo prevede sette scene, per ognuna delle quali sono disponibili 20 piedi di pellicola Ogni scena può essere girata un massimo di 5 volte, il che vuol dire, in pratica, che sono ammesse un massimo di cinque cadute se al termine della pellicola si è ancora in piedi si passa alla scena successiva, che naturalmente presenta difficoltà maggiori

Il capitale messo a disposizione dal produttore però è limitato, e se lo stuntman è lento si rischia la bancarotta

Per quello che riguarda grafia ed effetti sonori siamo su eccellenti livelli, in particolare lo spostamento delle sprite avviene con eccezionale fluidità

I controlli sono semplicissimi, tanto che il programmatore non ha previsto l'uso di un joystick, e si limitano ai tre tasti di destra, sinistra e salto

M B

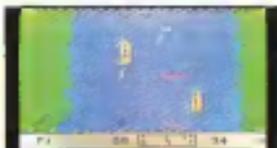
Produttore
Elite Systems Ltd
55 Bradford St
Walsall (GB)

Activision
RIVER RAID
MSX

Nonostante i dubbi e le perplessità di alcuni (principalmente la concorrenza), lo standard MSX sembra avviato con decisione alla conquista di una buona fetta del mercato degli home ad 8 bit, ne è prova la rapidità con la quale le principali case produttrici di videogame hanno iniziato la conversione per MSX dei loro titoli.

Tra i grandi nomi che hanno deciso di dedicarsi anche all'MSX troviamo anche l'Activision, della quale vi presentiamo questo River Raid, firmato dalla programmatrice Carol Shaw

Vediamone la trama alla guida di un caccia il giocatore intraprende una missione suicida che consiste nel risalire un fiume con lo scopo di infliggere il maggior numero possibile di danni al nemico. Lungo il fiume si trovano una serie di mezzi di tutti i tipi navi, elicotteri, aerostati (in guerra tutto va bene) che vanno distrutti con i cannoncini del proprio aereo, badando naturalmente a non farsi abbattere a propria volta Il carburante a disposizione non è molto, e durante la missione occorre rifornirsi spesso passando sopra i depositi, identificati dalla scritta FUEL La grafica ed il suono sono di buon livello e fanno meritare a River Raid un'ampia promozione, tuttavia da un MSX è lecito attendersi ancora di meglio

M B

Produttore. *Activision International*
Mountain View CA 94039 USA
Distributore per l'Italia:
Misa Traiding s r l
Centro Direzionale Milanofiori
Strada 7 Palazzo T1 20089 Rozzano (MI)

Chi di voi non ricorda la canzone Thriller, di Michael Jackson? È stato un successo della scorsa annata musicale, ed è diventato anche la videocassetta più venduta nel mondo, con centinaia di migliaia di copie vendute le immagini mostravano non solo una long version del favoloso filmato (costato, solo lui!, un milione di dollari per cinque minuti), ma anche svariati momenti del trucco degli attori che assecondavano Jackson

Poiché è fondamentale per la comprensione del gioco, vi ricordo per sommi capi la sceneggiatura del filmato, cui Chiller si rifà dichiaratamente Mentre il

nostro fa la corte ad una graziosa ragazzotta, esce fuori la luna piena, e il nostro viene assalito da tremiti - transilvani nel frattempo la passeggiata ha portato i due davanti ad un cimitero, in cui i morti risuscitano sotto forma di forzutissimi ed invincibili zombie, che assalgono la coppia, circondandola Nel frattempo lo spasimante Jackson si è completamente trasformato in un uomo lupo, con tanto di unghie, pelo foltissimo tipo Lucio Dalla, zanne elefantine e occhi gialli, gli effetti speciali hanno vinto il primo premio nell'assegnazione degli annuali oscar del videoclip, che si svolgono negli Usa Dopo un terrificante balletto con questi mostri, la ragazza fugge in una casa e viene lì raggiunta dagli zombie, che cercano di entrare in casa sfasciando le pareti il lieto fine è rappresentato dalla venuta del nostro eroe, che accendendo le luci scaccia gli incubi della fanciulla

Il gioco entra in azione con la fanciulla prigioniera dei mostri il nostro eroe deve liberarla nel tempo consentitogli dall'energia di cui dispone, che aumenta in prossimità di alcune tombe e diminuisce sensibilmente al contatto delle mostruose creature della notte Quattro gli schermi da attraversare, fino all'ultimo, che mostra il castello degli spettri in una delle sue stanze è nascosta la fanciulla, e voi dovete trovarla Ma adesso viene il bello, poiché con lei dovete tornare indietro, fino al punto di partenza - l'ingresso del cimitero

Dal punto di vista tecnico il gioco è completo il caricamento è allietato da un'ottima schermata in alta risoluzione, che mostra il nostro quasi trasformato in

licantropo e un'atterrita fanciulla (per l'occasione sono entrambi di pelle bianca) Si può giocare sia da tastiera che con il joystick, e le istruzioni sono in inglese, nonostante sul retro del cartoncino ci siano alcune frasi ad effetto stampate in diverse lingue, tra cui l'italiano Molto divertente la musica, anch'essa ispirata a Thriller il giro di basso della frase musicale principale è eseguito molto bene, e la melodia si basa sulle tre-quattro note che accompagnano Jackson mentre canta il nome della canzone

Chiller appartiene alla nuova serie di giochi Mastertronic per il 64, che si preannuncia densa di interessanti sviluppi

L S

Produttore:
Mastertronic Via Staurenghi 31
21100 Varese
Prezzo: *lire 7900 IVA inclusa*

Si tratta dell'ennesimo gioco spaziale per il Commodore 64, ben studiato non tanto nel gioco in sé, che non presenta particolarità di rilievo, bensì nella struttura generale caricamento veloce Data Jet, schermata introduttiva in alta risoluzione, più livelli di gioco, controllo da Joystick o da tastiera, e persino uno o due giocatori, possibilità troppo spesso trascurata dai programmatori, che vengono così a togliere al giocatore la sfida con l'avversario umano

Avete 5 vite, e dovete conservarle passando attraverso i 5 livelli di gioco predisposti da Mark McKee, che come al soli-

to è il realizzatore dei giochi della Channel 8 Software Se si compie tutto il percorso si guadagna una vita i vostri avversari sono inizialmente dei serpentoni, poi uno sbarramento e dei razzi che vi inseguono, quindi dei castelli e degli animali alati, al 4 livello abbiamo l'aeronautica, degli UB40 e dei carri armati, ed infine tutta una serie di improbabili creature Dobbiamo dire che la particolare gestione della grafica e degli sprite fa sparire parte del paesaggio al momento di bloccare il computer per poter fare le foto, che così appaiono meno movimentate della realtà

In definitiva si tratta di un gioco della fascia media, reso un po' più interessante dalla versatilità che offre all'utente

L S.

Produttore:
Channel 8 Software
51 Fishergate
Preston, Lancashire PR1 8BH GB

Finalmente un gioco diverso dal solito, con grafica interessante e al contempo abbastanza godibile per gli adulti ed educativo per i più piccoli. Loco è un gioco a percorso non difficile da padroneggiare, bastano poche ore, ma comunque interessante per lungo tempo. Loco è una locomotiva che parte dalla sua stazione per compiere un lungo viaggio. Molte le insidie: aerei e dirigibili la bombardano continuamente, mentre la linea è disseminata di carrelli vaganti, inoltre c'è una limitata disponibilità di benzina (fuel), che va continuamente ricercata nel percorso. Contro i nemici potete esplodere due tipi di colpi: con il vapore e con un cannoncino anteriore per i carrelli. Lo schermo è suddiviso in due parti, che scrollano lateralmente a differente velocità per mantenere un certo sincronismo: quella superiore contiene lo sfondo, ovvero montagne, prati, etc. con la strada ferrata, e gli sprite con la locomotiva (formata da 4 pezzi) e gli eventuali ostacoli; quella inferiore, invece, contiene una vasta parte della mappa del percorso ferroviario, densa di scambi, ponti, etc. ed ha lo scopo di mostrare l'ubicazione dei vari aerei, dirigibili, carrelli e depositi di carburante. In questo modo si possono evitare i problemi al loro sopraggiungere, per esempio cannoneggiando i carrelli ovvero lanciando cortine fumogene sugli aerei e sulle loro bombe prima che giungano su di voi, sfruttando il fatto che il fumo, in quota, devia per effetto del vento. Aerei e dirigibili vanno ad altezze diverse, e il vapore che lanciate può essere regolato dal tempo di pressione dell'apposito comando sul joystick. Tutti i comandi, abbastanza articolati ma sufficientemente intuitivi, sono spiegati graficamente sul contenitore del gioco.

Avete a disposizione tre vite, e ne guadagnate una ogni 10.000 punti, se passate 5 volte per la stazione passate al livello successivo (ce ne sono 5).

Il caricamento veloce (Superfast Loading) dura solo un minuto e mezzo. Anche la musica è particolarmente curata, e cerca di riprodurre un buon numero di strumenti reali: gli autori, Crowter e Dalglish, hanno fatto un ottimo lavoro, che magari non soddisferà appieno i più bravi, ma che senza dubbio è ben studiato dal punto di vista estetico e validissimo per bambini e ragazzi, come dice la stessa casa (età: da 9 in su). *L.S.*

Produttore:
Alligata
1 Orange Street
Sheffield S1 4DW - GB

Vi trovate alla guida di un elicottero, distaccato su una portaerei nel bel mezzo dell'oceano. La vostra missione è salvare dei superstiti che si trovano sulla terraferma, in una zona protetta dalla base nemica, che ospita un reattore nucleare; per farlo dovrete evitare la contraerea nemica, situata su alcune isolette, e i vari dirigibili che controllano lo spazio aereo, oltre che annientare (momentaneamente) la base nemica, caricare gli uomini e rifare il percorso, all'indietro. Per far ciò avete a disposizione un certo tempo, indicato dal carburante (fuel) residuo. Il commento sonoro è sufficientemente realistico, soprattutto nel rumore dell'elicottero; il caricamento veloce assicura un'attesa breve.

Il programmatore, Val Franco (il nome tradisce l'origine italiana) ha predisposto 5 avventure, delle quali le prime 4 possono essere scelte premendo i tasti numerici da 1 a 4, mentre la quinta, senz'altro la più affascinante, può essere giocata solo portando a compimento la quarta (cosa che finora non ci è riuscita).

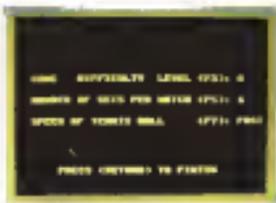
Il nostro giudizio su Blue Thunder è quindi sostanzialmente positivo, anche se avremmo preferito degli effetti sonori più corposi e la possibilità di avere due giocatori nella stessa partita, opzione che, a nostro avviso, non dovrebbe mai mancare. *L.S.*

Produttore:
Richard Wilcox Software
Foundry Business Systems
2 Station Road
Walsall, West Midlands WS2 0JZ - GB

Merlin Software
WIMBLEDON
Commodore 64

Diversi mesi fa, nel reportage dalla 5a Commodore Exhibition di Londra, vi parlammo della Merlin Software, basata sul lavoro di un giovanissimo inglese di Manchester, Jason Perkins, che — allora appena sedicenne — aveva realizzato svariati giochi per il 64 (Orange Squash, Pixie Pete, etc), ma soprattutto mostrava orgoglioso il suo super programma. Wimbledon 64, un gioco di simulazione del tennis, il primo (e per noi anche unico) che giri sul Commodore 64, e che doveva ancora essere prodotto in grande serie. Eccolo qui, nello splendore del technicolor!

Il gioco in se e molto ben fatto, e la grafica e curata in molti particolari: l'ombra della pallina, ad esempio, assume dimensioni diverse a seconda della distanza dal suolo, così come pure la racchetta, che vista frontalmente e più grande che non vista di tre quarti. Come al solito in giochi di tennis, il lancio della pallina viene simulato dal computer, che la fa muovere perpendicolarmente al suolo: due i modi di battuta, o tipo smash (la stessa impugnatura della schiacciata) ovvero da principiante, dal basso (!); l'angolo viene dato muovendo lateralmente il joystick, mentre per ogni colpo la forza e proporzionale al tempo di premuta del tasto rosso. Il joystick controlla il moto del giocatore, ma premendo il tasto rosso si gestisce la racchetta, che quindi — almeno al livello iniziale — si muove in tre direzioni (diritto, rovescio e smash) con forza variabile nel modo descritto. La pallina va colpita abbastanza vicino alla racchetta, anche se non c'e cambiamento in dipendenza del punto dell'incordatura sfruttato.

Familiarizzare con Wimbledon non è per nulla facile, a partire dal servizio, passando per gli scambi da fondo campo, per giungere alle schiacciate a rete, per questo il programmatore ha inserito molteplici varianti di gioco, con 4 livelli e due velocita per la palla, oltre che a numero di set variabile (da 1 a 3). Con il livello varia anche il numero di colpi resi possibili dal gioco. Come saprete, nel vero Wimbledon e stato recentemente accettato il tie-break, per limitare la durata dei set: questa simulazione ne tiene conto, ma sull'otto pari si limita a far giocare un normale diciassettesimo gioco (non un tie-break) assegnando la partita al vincitore dell'ultima frazione.

Alcune scelte discutibili hanno portato ad una difficile determinazione del punto di caduta della palla in prossimita delle linee, che generalmente vengono assegnate buone, e il commento sonoro, veramente esiguo dato che si limita a due suoni, peraltro improbabili, quando la palla cade o viene colpita dalla racchetta. L'unico aspetto negativo per la versatilità del gioco è l'impossibilita di giocare contro il computer, per cui servono assolutamente due joystick e due giocatori! Crediamo che questo fatto possa costituire un freno alla diffusione di Wimbledon ai livelli del Soccer, ad esempio.

L S

Produttore:
Merlin Software
Business & Technology Centre
Bessemer Drive Stevenage
Herts SG1 2XD - GB

Commodore Plus/4

di Leo Sorge

Sono passati svariati anni da quando la Commodore annunciò l'intenzione di darsi ai personal computer veri. La storia partì nel 1982, all'alba dell'introduzione in Italia del C 64, che allora costava 975 mila lire più IVA, ovvero quanto costò a chi scrive (che peraltro trovò la cosa conveniente, rispetto alle 475.000 più IVA che era costato, l'anno prima, il VIC 20). Dagli USA rimbalzò la notizia secondo cui la casa aveva già pronti i successori del 64, chiamati 264 e 364, ancora più completi del predecessore, con un Basic all'altezza dell'hardware, software integrato e sintesi vocale per il secondo: la stessa casa ne ritardava il lancio cercando di vendere più pezzi possibile del 64, il cui clamoroso successo li aveva colti impreparati: questo home, infatti, era solo una prova generale!

I due fratelli, il 264 e il 364, presentati negli States al CES del gennaio '84, e mai lanciati negli USA, non conobbero mai l'Inghilterra — sempre la prima nazione europea nel campo dei computer — e furono visti all'Hannover Fair dello stesso anno: dopotutto, buona parte dei Commodore europei vengono assemblati in Germania. Questa apparizione, però, era solo fittizia: già si sapeva che non era quella la veste definitiva con cui sarebbero stati lanciati sul mercato: ciononostante alcune riviste italiane apparvero con servizi sull'argomento, o per allettare i meno smaliziati oppure in un improvviso e sospetto rigurgito di fiducia nella Commodore Europea.

Ma vediamo com'è questo Plus 4 in rapporto alle molteplici situazioni in cui si trova immischiato. Innanzitutto è un Commodore, e segue in parte la politica della casa, per cui i nuovi modelli sono sempre incompatibili con i vecchi — a parte il Basic all'osso — e per di più sia l'hardware — processore, memorie via — che il software — Basic, interfacce — è tutt'altro che standard. Poi non è, come i predecessori, un home computer, ma molto più vicino al personal, e ciò nonostante le limitazioni nel video, sempre a 40 colonne, e l'incompatibilità con i sistemi operativi standard (DOS, CP/M, ecc.), con del software su Rom, di buona fattura e agevole uso. La struttura interna, però, trasuda la vecchia esperienza nel campo hobbistico, e allora il Basic è valido, e sfrutta adeguatamente l'hardware di base, oltre a gestire in modo assai esteso i dischi, semplici e doppi, e a offrire un assembler-disassembler ed editor per la programmazione in linguaggio macchina.

Chi però si aspettava che il Plus 4 fosse anche un home potente come il 64 sarà certamente deluso: la mancanza degli sprite, e di qualsiasi forma di sintesi musicale, lascia davvero stupefatto chi non capisca che questo computer è un nuovo capitolo della storia Commodore, ancora troppo vicina al passato per potersene distaccare completamente.

Una cosa da dire in apertura e non ripetere più è che non abbiamo trovato nessun accenno relativamente ad un uso del software integrato tramite cassetta: il sistema si basa decisamente sui dischi, e presto dovrebbero essere lanciati sul mercato dei drive veloci, dei doppi dischi ed altre periferiche di qualità, così come stampanti silenziose, bidirezionali e anche a colori sono oramai nell'aria.

In definitiva il Plus 4 rappresenta l'ennesi-

ma sfida Commodore al mercato dato che le grosse cifre si stanno spostando dagli home ai personal la dinamica casa americana sta cercando di crearsi una nuova verginità nel settore di maggior richiamo. L'eventuale successo dipenderà sua dalle caratteristiche tecniche che dal combinarsi di fattori di mercato se per i secondi non possiamo entrare in merito alla questione, cerchiamo almeno di stabilire se ci sono i presupposti tecnici.

L'estetica

Per il lancio di un prodotto di tipo nuovo, come questo, che si immette nel settore professionale, occorreva una nuova impostazione, anche per dissociarsi dall'immagine precedente, quella di home computer partita con il Vic, rafforzatasi con il 64 e mantenuta con il C 16 (la cui prova, vi ricordiamo, è stata pubblicata sul numero scorso) Questi principi hanno portato ad un mobile di dimensioni appena più grandi della tastiera stessa, tra l'altro direttamente derivata da quella del portatile Executive, con i tasti funzione di piccole dimensioni posti frontalmente, e il controllo del cursore affidato a 4 comodi tasti a freccia, messi in basso a destra I tasti bianchi risal-

Costruttore:
Commodore Electronics
1 Hunters Road Weldon Corby
Northants NN17 1QX
Gran Bretagna
Distributore per l'Italia:
Commodore Italiana
Via F.lli Gracchi 48
20092 Cinisello Balsamo (MI)
Prezzo:
Lire 845.000 + IVA

Il pannello posteriore del computer Da sinistra verso destra vediamo la strana presa per l alimentazione il DIN del bus seriale Commodore e l'insolita connessione per il registratore a cassette Segue la user port, gestita direttamente come interfaccia RS-232C (ma a livelli TTL) e la slot di espansione per le cartucce Chiudono la serie i due ingressi per joystick anch'essi strani e l uscita per monitor

tano parecchio sullo sfondo grigio molto scuro, conferendo un tocco di austerità appena attenuato dalle bande colorate poste al di sopra dei tasti funzione e dalle fessure d'areazione introdotte nella parte superiore del mobile.

Veniamo adesso ai collegamenti e ai controlli Sul lato destro trova posto un comodo tasto di reset, che purtroppo cancella il contenuto della memoria (in effetti rimette a posto i puntatori) Al suo fianco troviamo l'interruttore d'alimentazione. Sul lato posteriore ci sono moltissime connessioni, e di strano tipo con riferimento alla foto, andando da sinistra verso destra, abbiamo la presa per le varie alimentazioni (di forma quadrata, insolita), quella dell'interfaccia seriale Commodore, che gestisce le periferiche, il connettore per la cassetta, uno stranissimo oggetto a 7 piedini, che diventano 8 nelle due prese per joystick, poste sull'estrema destra — sempre vedendolo dal posteriore — e affiancata a quella per il monitor; il pin per il televisore è invece posto sull'altro lato.

Le due parti più interessanti del pannello posteriore sono i due slot per le cartucce e la user port; la prima consta di uno slot da 25+25 piedini a passo ridotto (1,56 mm), mentre la porta utente è un pettine da 12+12 piste di passo 4,56 mm, servite sia dall'8551 della RS-232 che da uno dei due ACIA (chip specializzati per l'I/O) 6529

La completa tastiera del Plus 4 Notare i tasti funzione sottili in alto a sinistra e quelli che controllano il cursore configurati come frecce a croce, in basso a destra

Il rucolhardware del Plus 4 comprendente il banco di Ram (in basso a destra), il firmware (i 5 chip paralleli al centro), il chip audio-video inscatolato in una gabbia metallica e l'elettronica di gestione dell'RS-232C

Hardware

L'architettura del Plus 4, escludendo il software su Rom, è decisamente inferiore a quella del 64. Infatti tutte quelle funzioni grafiche e sonore che avevano rappresentato la forza del predecessore, sono ora ridotte all'osso e compattate in un unico chip siglato 8360, in pratica una ULA che genera il video e il suono. Il microprocessore si chiama 8501, ed è una versione aggiornata del 6502, con cui è soft compatibile: il monitor incluso nella macchina, infatti, si basa sui codici mnemonici di quel microprocessore. Oltre a questi due, nell'interno del microprocessore appaiono svariati altri circuiti integrati: di questi, quelli di base sono quelli siglati 05, 01 e 02, che dovrebbero essere le Rom con il Basic, il sistema operativo e il monitor per LM. Rimangono i due grossi chip che contengono i 4 programmi applicativi: sono altri due siglati 01 — guardando frontalmente l'unica piastra sono il terzo e il quarto da sinistra —. Il banco di Ram si trova in basso a destra, ed è composto da 8 Oki 3764, da 64Kbit l'una per un totale di 64Kbyte. È interessante la gestione del Basic, che viene visto quasi come una periferica: i 64K gestiti dal microprocessore sono infatti interamente di Ram, eccettuati 4K della pagina video e dei salti alle routine del Basic, presumibilmente eseguite tramite selezione software tra banchi di Rom e di Ram.

Rimane da dire qualcosa sul chip 8551, che appare alla destra del microprocessore. Sia il 64 che il Vic avevano una porta RS-232C per comunicazioni con modem (per le stampanti era molto difficile impiegarla), la cui caratteristica era di avere i livelli d'uscita ai valori TTL e invertiti. Queste interfacce vengono generalmente gestite da appositi circuiti integrati, ma in quel caso la Mos, che costruisce i chip per la Commodore (cui appartiene al 100%) non fece in tempo a realizzare quello adatto, che sarebbe dovuto essere un 6551. La Commodore simulò via software il circuito mancante mantenendone invariata la gestione tramite registri: ciò però limitò la massima velocità di trasmissione raggiungibile. In questo nuovo computer la porta RS-232 è gestita da hardware, tramite il fatidico 8551, che tollera la massima velocità di trasmissione del sistema ovvero 19200 baud: è ovvio che tale possibilità è ben sfruttata solo tramite il linguaggio macchina.

Il parallelismo tra la struttura del 64 e quella del Plus 4 è completato dai due 6529b, che sostituiscono i 6526 del 64, non mantenendone le dimensioni: uno viene usato per la tastiera, l'altro principalmente per gestire la User Port, ma tali distinzioni non sono così nette come sembrerebbe, dato che la versatilità di questi circuiti ha senza dubbio portato ad intrecciarne i compiti. Tutte le mappe di memoria sono riportate in forma tabellare, a pagina 200 del manuale inglese, la mappatura dei registri si trova a pag. 199.

Il Basic

Il linguaggio implementato sul Plus 4 è il medesimo del C 16, l'home computer lanciato dalla Commodore: si tratta di un 3.0 dal punto di vista della programmazione, particolarmente arricchito nei comandi per disco e per l'hardware di contorno come joystick, suono e affini, quindi chiamato 3.5.

La programmazione in senso stretto si è notevolmente strutturata, sia per la presenza dell'ELSE dopo l'IF-THEN che per la possibilità di effettuare cicli più intelligenti come quelli consentiti dalle strutture LOOP-UNTIL e WHILE, da tempo introdotte nei Basic tendenti a rivaleggiare con il Pascal. Un'aggiunta eccellente per poter preparare del serio software in Basic senza pesanti routine in linguaggio macchina, è il completo tool per la ricerca degli errori e la diramazione del programma in conseguenza del tipo. Gli errori sono tutti numerati, e il codice è contenuto nella variabile dedicata ER, mentre la stringa del messaggio corrispondente si trova nell'altra variabile ERR$. Quando in una linea del program-

Tutto il software del computer. I cinque chip paralleli contengono il firmware, cioè wordprocessor, spreadsheet, business graphics, file manager, monitor LM, Basic 3.5 e sistema operativo: il firmware è il software su Rom contenuta nella macchina

L'8360: quasi un ULA visto che gestisce al contempo il video e il sonoro. Per evitare interferenze in alta frequenza è usualmente protetto da una gabbia metallica, cui è anche accoppiato termicamente per migliorare la dissipazione: si noti infatti il residuo di silicone (la macchia bianca), che tiene l'aletta fissata al chip

L'8551 e il 6526, entrambi di produzione MOS, gestiscono la RS-232C come al solito a livelli TTL anziché quelli dello standard

Un esempio di wordprocessor. Osservate il display a 37 caratteri per riga, nonostante le linee siano di 77 battute

Usando tutto lo schermo, la spreadsheet mostra 13 righe di 3 colonne

Uscita del programma grafico che, a partire da una riga dello spreadsheet, genera un istogramma verticale che mette nelle prime 25 righe del wordprocessor

ma si può verificare un errore, in testa si mette la parola TRAP <numero di linea>, subito dopo la linea incriminata si pone un salto ad opportuna routine che tenga conto dell'errore intervenuto (magari verificando, tramite ER ed ERR$, di quale si tratti); infine si pone una RESUME <linea di programma>, che nel nostro esempio, avendo noi messo la linea di salto subito dopo quella con l'errore, si riduce ad un RESUME NEXT. Per meglio vedere cosa accade facciamo un esempio pratico, invece dell'usuale

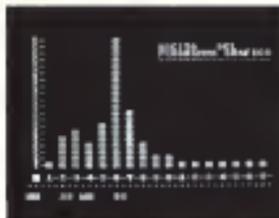
```
110 <linee con l'eventuale errore>
120 ON ERROR GOTO <routine di controllo>
```

dovremo porre

```
100 TRAP 110
110 <linee con l'eventuale errore>
120 GOTO <routine di controllo>
130 RESUME NEXT
```

di grande utilità è anche il gruppo di istruzioni per la stesura e la correzione di un programma, ovvero i soliti AUTO, DELETE e RENUMBER, seguiti dai TRON e TROFF, è purtroppo ancora assente una routine di OLD, della quale si sente assai la mancanza data anche l'assenza del comodo tasto di Restore, che sul 64 risolveva parecchi problemi, né basta il tasto di Reset, che purtroppo cancella i programmi (a meno di andarseli a ripescare con il monitor, possibilità questa semplice e tutt'altro che remota).

La formattazione dei programmi si presenta in modo eccellente, nella pagina testo abbiamo un Print Using dalle infinite possibilità, che consente non solo la normale gestione delle scritte, ma anche la conversione tra programmi delle varie interpunzioni ed altre interessanti cosine' vi diciamo soltanto che sul manuale, peraltro piuttosto denso, la trattazione di riferimento occupa ben 4 pagine. Le possibilità di ingentilire i vostri programmi in modo semplice si estende anche alle varie pagine grafiche, dove ben due comandi, CHAR e TEXT, consentono di apporre scritte sui grafici o disegni in alta risoluzione.

Anche le stringhe sono state rese più potenti, tramite l'introduzione del comando INSTR, che accetta in ingresso i nomi di due stringhe (ed eventualmente una posizione di partenza) per restituire la posizione della stringa 2 nella 1: per meglio capire, avendo

```
A$ = "microcomputer"
```

e

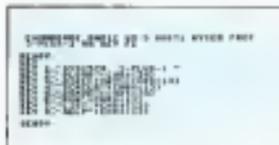
```
B$ = "comp"
```

impostando una linea del tipo

```
C = INSTR (A$, B$)
PRINT C
```

otterremo il valore 6, poiché comp è contenuto in microcomputer, e la sua prima lettera parte dalla sesta posizione.

Grafica e suono

Vediamo cosa è successo a quella che era la parte più interessante del Commodore 64. Se prima c'erano due super-chip specializzati, il VIC-II 6567 per la grafica e il SID 6581 per il suono, adesso c'è un unico chip l'8360, che come abbiamo visto è ingabbiato (queste precauzioni non fanno mai male). Prima sorpresa: mancano gli

All'accensione i tasti funzione realizzano un primo one-stroke Basic: ecco cosa scrivono i vari tasti

sprite! Seconda sorpresa: manca la possibilità di definire i parametri dell'onda sonora (l'ADSR e il resto)! Sbaglierebbe grossolanamente la propria analisi chi pero andasse a valutare questi fatti nell'ottica del l'home computer: il Plus 4 NON È assolutamente un prodotto da inserirsi in quella fascia, e le sue possibilità software sono assai estese. Basti pensare alla grafica, che offre ben 5 modi:

0 testo 40 × 25
1 alta risoluzione 320 × 200
  (1 colore + 1 di sfondo)
2 alta risoluzione 288 × 200
  + 4 linee di testo
3 grafica a colori 160 × 200
  con 4 colori
4 grafica a colori 144 × 200
  a 4 colori e 4 linee di testo

La pagina grafica, che occupa 10K, viene allocata a partire dalla locazione 4096, da dove usualmente parte la Ram del listato Basic. Quest'ultimo va a mettersi dopo l'alta risoluzione, e non viene eliminata neanche tornando al solito modo testo, ma solo digitando GRAPHIC CLR.

Particolarmente interessante è lo schermo diviso (in inglese "split screen", si trova spesso sul manuale), che si basa su un'accorta gestione software della routine di generazione dell'immagine video. In pagina testo è comunque possibile riempire lo schermo con opportune immagini e poi definire una finestra di tipo testo, ponendosi nel luogo di partenza e premendo prima ESC e poi B, per poi andare nell'ultima locazione e digitare ESC T, in quella finestra si possono far scorrere da programma altre immagini o testi.

Un'ulteriore, eccellente possibilità di questo Basic 3.5 è sempre relativa all'alta risoluzione, e ci consente di gestire parte delle immagini in alta risoluzione come se fossero stringhe: si può infatti usare la

```
SSHAPE <variabile$> a1 b1, a2, b2
```

(Save Shape, memorizza una forma) ove i 4 parametri letterali sono le coordinate dei punti estremi in alto a sinistra e in basso a destra; omettendo la seconda coppia si intende il vertice destro. L'istruzione contraria è GSHAPE (Get Shape, prendi l'immagine), che può rimettere i punti in posizioni diverse da quelle originali — sempre specificando il punto di inizio — sia com'era, che invertita, che facendo l'OR, l'AND e l'XOR con quello che si trova sotto. L'unica altra volta che abbiamo trovato questa funzione è stato sul Dragon, che usava la parola GET. È abbastanza difficile, anche se non impossibile e comunque assai lento da eseguirsi, realizzare un programma Basic che sfrutti la SSHAPE per generare qualcosa che somigli a degli sprite, va comunque ricordato che essendo 255 il massimo numero di caratteri di una stringa (in Commodore Basic), dato che gli ultimi 4 byte contengono le coordinate di riferimento, in ogni stringa entra una zona di 251 × 8 punti, pensando ad ogni punto come un bit, questo conto vale in alta risoluzione, e va diviso per due in modo multicolore.

Diamo ora uno sguardo al suono. Ab-

biamo già detto che manca qualsiasi accenno di sintesi musicale, e per di più disponiamo di due sole voci, la prima di solo suono con il codice 1, la seconda essendo selezionabile sia come suono — codice 2 — che come rumore — codice 3 — Due i comandi che gestiscono questa caratteristica. VOL <numero>, che accetta valori da 0 a 8, e il cui senso è di banale interpretazione, e SOUND, di sintassi

SOUND a, b, c

dove a indica il codice della voce, che come visto può variare da 1 a 3, b determina la frequenza, con un valore da 0 a 1023, c determina la durata in sessantesimi di secondo, e può valere da un minimo di 1 ad un massimo di 65535 Molte sono le annotazioni da farsi sul parametro b. Innanzitutto le scale indicate sul testo per riprodurre le solite temperate sono sbagliate, così come pure risultano approssimative le espressioni indicate per determinare la frequenza da generare. Inoltre il valore massimo di tale parametro non si ottiene, come si potrebbe pensare e come è indicato sul manuale, con il valore massimo accettato, cioè 1023, ma con una o due unità in meno, se però 1022 risulta inudibile, 1023 equivale — sul nostro modello — alla frequenza ottenuta impostando circa 100 Va anche precisato che l'algoritmo che genera i numeri casuali da cui parte la base dei rumori della seconda voce si ripete in modo abbastanza ciclico, per cui quello che dovrebbe essere uno sgraziato effetto aperiodico, dopo un certo valore che per noi è superiore a 1000 diventa uno strano rumore elettronico, ma periodico e quindi riconosciuto dall'orecchio come suono, anche se di frequenza opportunamente inferiore (più o meno un terzo) di quella teorica, anche il generatore di rumore, per b = 1023, offre lo stesso risultato offerto dal parametro 100.

Un esempio di uso del monitor: un breve disassemblato

Il due grossi manuali del Basic e del software applicativo il secondo è organizzato come un corso per principianti e completato da guide di riferimento per i più esperti

Il resto del Basic

La prima cosa che balza all'occhio e la presenza sulla tastiera di alcune nuove scritte Innanzitutto in alto a sinistra campeggia un tasto di ESCape, che in questo computer non e altro che un ulteriore modo per sfruttare i tasti già esistenti Come abbiamo già visto, ESC T generano una finestra in pagina testo, molte lettere (I,J,K), precedute da ESC, lavorano ampliando il già eccellente editor di schermo del Plus 4, ma a parte va citata la funzione di inserimento offerta da ESC A, che se attivata permette di inserire nuove scritte in una riga di listato, senza per questo dover usare il solito Shift + Inst/Del Veramente utile anche lo scroll nei due sensi verticali, alto e basso, offerto dai tasti W e V, sempre se preceduti dall'Escape

Una sezione a parte dell'articolo se l'è senz'altro meritata il set di istruzioni dedicate al disco, e ne avremmo parlato diffusamente se solo avessimo potuto disporre di qualcuna delle periferiche di nuovo tipo presentate in parte alla scorsa Commodore Exhibition ci riferiamo al disco veloce 481 e ai nuovi singoli doppi ad alta velocità e capacità Mancando di riferimenti dobbiamo limitarci a far rilevare che tutte le funzioni precedentemente raggiungibili solo tramite complesse operazioni con il disco (parliamo del Vic e del 64) sono adesso ottenibili direttamente dal Basic Così HEADER (intestazione) formatta un dischetto, DIRECTORY mostra sullo schermo il catalogo del disco, COLLECT riorganizza il disco sostituendo la vecchia Validate disponibile sul 1541, BACKUP copia un dischetto non protetto, COPY fa lo stesso su file singoli, DLOAD e DSAVE caricano e salvano usando il disco e così via, ovviamente sono ancora valide tutte le alchimie in auge sul 64, per cui un programma si carica dal disco con un Load '<nome>', 8, e allora il directory, che e un programma di nome '$', si mette in memoria (ma distruggendo quello presente) come un qualunque file, e Dio ci scampi dalle operazioni di formattazione, copia, cancellazione di un file!

In chiusura accenniamo alla possibilità di gestire i tasti funzione assegnando loro delle stringhe — comprendenti anche eventuali codici di controllo — che vengono stampate automaticamente al momento

Il directory ottenuto dal ... dagli altri programmi notate che il numero di blocchi viene dato in tre cifre

della pressione dei tasti la parola chiave e, neanche a dirlo, KEY. Ovviamente all'accensione il computer e già programmato con una stringa per ogni tasto, compresi quelli duplicati premendo lo Shift, sempre KEY, ma senza alcuna specificazione, fornisce l'elenco delle stringhe

Il monitor

Monitor vuol dire letteralmente 'controllo', per cui le funzioni di un tale programma dovrebbero essere alquanto limitate. Il fatto di dialogare direttamente con il microprocessore e i suoi chip periferici ha però portato ad espandere le funzioni di questo programma, che in effetti attualmente corrisponde ad un assemblatore, ad un disassemblatore e ad un piccolo editore di programmi scritti in linguaggio macchina come avrete capito, si tratta di un tool indispensabile per i programmatori evoluti.

Vediamo le principali possibilità offerte da questo programma, che la Commodore chiama Tedmon

A assembla una linea in LM del 6502,
D disassembla,
M mostra la memoria, e la cambia direttamente modificando lo schermo,
S salva su nastro o su disco,
L carica da nastro o da disco,
X ritorna al Basic,
G (go = vai) esegue un programma,
e molte altre, tra cui F (fill) per riempire la memoria con un dato carattere, oppure T (transfer) che trasferisce altrove un blocco di memoria, V che verifica il caricamento o il salvataggio del programma, H (hunt = scova) che setaccia la memoria alla ricerca di un particolare byte, e qualche altra cosa per un totale di 13 comandi e 3 operatori La maggioranza dei programmatori in 6502 avrà certamente riconosciuto gli usuali comandi di programmi analoghi, ormai più o meno standard, il tool è decisamente potente, ed apre nuove frontiere ai commodoriani, in quanto risiede direttamente in Rom

Il software

Uno dei principali punti di forza del Plus

Si possono visualizzare contemporaneamente lo ss e il wp con 12 linee ciascuno lo stato del wp si trova comunque nell'ultima riga È possibile trasferire al wp dei dati contenuti nello ss

Il registratore a cassette più sottile dei precedenti Il connettore usato è davvero strano!

4 e senza dubbio la presenza di 4 programmi applicativi inclusi nella macchina — e nel suo prezzo [1] — sotto forma di Rom Il software proposto viene definito integrato, ma visto l'incredibile uso che oggigiorno si fa di questo termine, sarà meglio definirne i confini in modo preciso Abbiamo a disposizione tre programmi effettivi, un wordprocessor, uno spreadsheet e un file manager, e in più una routine che genera istogrammi da questa disposizione presumibilmente proviene il nome dell'intero pacchetto, chiamato 3+1 D'ora in poi, per brevità, chiameremo i programmi con le abbreviazioni wp, ss, fm e bg, con ovvi significati

Il programma di base è il wp Si tratta di un foglio di 99 righe per 77 colonne, per un totale di circa 7600 battute Lo schermo mostra contemporaneamente 22 righe da 37 caratteri, ma uno scroll laterale completamente automatico gestisce il foglio intero. I comandi sono molteplici, e si accede loro andando in Command Mode, ovvero premendo contemporaneamente i tasti con il logo Commodore (che d'ora in avanti chiameremo CBM) e la C È previsto un completo set di istruzioni di formattazione della stampa, compreso un comodissimo linkfile <nome file> che dopo aver stampato il file in memoria carica e stampa quello di nome <nome file> Il controllo del disco è eccellente, dato che permette non solo il caricamento e la registrazione, ma anche la visione del directory del dischetto (opzione 'CA', catalog) ed altre cosine interessanti Rispetto a programmi dedicati, come il Superscript e l'Easy Script che girano sul Commodore 64, abbiamo una certa difficolta ad effettuare cambiamenti, dato che manca la comodissima opzione di inserimento diretto di righe o nuovo testo — soppiantata a favore di una macchiosa gestione di blocchi di testo da trasferirsi — così come pure la possibilità di cancellare selettivamente alcune zone dei documenti, a parte un DL- delete line

Lo spreadsheet e un tabellone di 50 righe per 17 colonne, e risulta quindi più che sufficiente per tutti gli usi domestici e di piccola contabilità È abbastanza versatile, e non abbiamo sentito alcuna mancanza rispetto a package decisamente più evoluti

A confronto le dimensioni del connettore del registratore (tra l'altro identica a quella del joystick) e una dei soliti DIN

come ad esempio il Calc Result, dato che sono presenti moltissime funzioni tipo MAX e MIN per mettere in una casella il valore massimo o minimo in un certo gruppo di altre caselle, ed inoltre possiamo effettuare una qualsiasi delle 4 operazioni, ripetutamente, tra varie caselle della stessa riga e colonna, o anche possiamo adattare una formula da una riga all'altra, senza dover riscriverla per intero È presente anche una certa versatilità di programmazione, dato che possiamo usare l'IF-THEN con molti operatori logici per determinare il verificarsi di una condizione tra due o più possibili Con lo ss si inizia a vedere l'integrazione del software tramite il comando HA (half=metà) e possibile dividere lo schermo in due parti a sviluppo orizzontale, richiamando in quella superiore 12 righe di wp, (con il comando tw, to wordprocessor), in modo da commentare i valori numerici del tabellone vedendoli nella meta inferiore dello schermo (a sole 7 righe). È ovviamente possibile trasferire al wp i dati contenuti nello spreadsheet, onde presentare documentazioni appropriate

In quest'ambito si inserisce la routine grafica, che disegna istogrammi relativi ad una qualsiasi delle righe dello ss, effettuando una scala automatica (ovviamente basandosi sul valore minore e su quello maggiore) e ponendo il risultato direttamente nel wordprocessor, a partire dalla locazione 0,0· nell'uso di questa routine bisognerà quindi porre notevole attenzione, onde evitare di cancellare del testo scritto in precedenza nelle righe iniziali

Il quarto programma del pacchetto è il fm, in pratica un piccolo data base particolarmente versato per realizzare agende telefoniche o mailing list. Per ogni record avete a disposizione 17 campi da 38 caratteri ciascuno, e in ogni file entrano al massimo 999 record I comandi del disco e della gestione comune (catalogo, memoria, ecc) sono gli stessi già visti per il wp, e valgono ovviamente anche per lo ss L'inserimento di dati si presenta decisamente confidenziale, ed offre alcune particolarità di rilievo quali il sort sia per lettere che per numeri, oltre che la stampa collegata al wp

In finale di paragrafo spendiamo alcune note sul manuale in inglese Noi l'abbiamo trovato completo e ben fatto, dato che si tratta di un corso di apprendimento basato su lezioni successive, più una guida di riferimento finale e qualche appendice Speriamo che le quasi 220 pagine vengano tradotte in modo opportuno, e che la qualità delle illustrazioni, appena sufficiente, non subisca ulteriori —e in questo caso, decisivi— peggioramenti

Conclusioni

L'architettura della macchina, nonostante sia semplificata rispetto a quella del C 64, rimane decisamente di qualità, la grafica e ben sfruttata e accessibile, e l'interfacciamento curato, rimane da chiedersi cosa ne sara dell'assistenza, dato che la Commodore usa circuiterie di sua esclusiva produzione Il software integrato, sul quale torneremo a breve scadenza dedicandogli un articolo, e di buona qualita, oltre che decisamente semplice da usare, ed e espressamente indirizzato alle esigenze del professionista che voglia gestire in proprio la sua attività

La buona qualita del Basic, e la quanto mai opportuna presenza del monitor per LM, rende facile prevedere una veloce apparizione di programmi validi sia per l'hobbista che per l'utente evoluto.

In definitiva riteniamo il Plus 4 un buon personal computer, già predisposto per l'eventuale apprendimento della programmazione sia Basic che in LM al prezzo attuale di 845 000 lire più IVA, con tutta probabilità destinato a subire un ulteriore ritocco verso il basso, e senza dubbio un buon investimento, tenendo presente che al prezzo della macchina va aggiunto quello di un floppy disc (il 1541 costa circa mezzo milione), quello di un monitor in bianco e nero (150-200 mila lire) e quello di una stampante Commodore (a partire da 650 000 lire circa) La parola passa ora al pubblico MC

Apricot è l'unica famiglia di personal computer con compatibilità totale. Tutte le macchine possono usare lo stesso software, sia scambiandosi i Floppy disks che collegandosi in rete. Le caratteristiche comuni alla famiglia sono: Microprocessore 16 BIT Intel 8086 - Sistema operativo MS DOS (opzionali CPM 86, Concurrent CPM, Concurrent DOS) - Memoria centrale standard 256 Kbyte - Tastiera ergonomica con sezione numerica separata - Trasferibilità dati e programmi da e per sistemi PC compatibili.

APRICOT PC

Tastiera con microscreen a cristalli liquidi. Video monocromatico ad alta risoluzione. Memoria di massa fino a 1 440 Kbyte. Memoria RAM espandibile a 768 Kbyte. Facile trasportabilità. Prezzo da lire 5 400 000

APRICOT Xi

Oltre ad avere tutte le caratteristiche dell'Apricot PC, dispone di Winchester incorporato da 5 a 10 Mb. Prezzo da lire 9 200 000

APRICOT PORTABLE

È il più potente portatile esistente sul mercato mondiale. Schermo a cristalli liquidi professionale con 25 linee da 80 caratteri. Unità disco da 720 000 caratteri. Dispositivo di riconoscimento di comandi vocali. Tastiera e mouse collegati a raggi infrarossi. Collegabilità con schermo a colori, stampante portatile e Winchester. Pesa meno di kg 6. Prezzo da lire 5 000 000

APRICOT F1

È il più economico della linea professionale. La tastiera comunica a raggi infrarossi. Può collegare il mouse e il video a colori. Floppy disk da 720 Kb. Esiste anche una versione semplificata per uso domestico e scolastico da 128 Kb e Floppy da 315 Kb. Prezzo da lire 2 800 000

POINT 32

Un sistema LAN (Local Area Network) che può collegare fino a 32 unità Apricot o Victor o PC-compatibili, con archivi in comune fino a 100 Mb su disco, e con sistema di salvataggio dati con cartucce a nastro da 200 Mb. Prezzo da lire 10.000.000

APRICOT. UNA FAM

Tutti i sistemi vengono corredati di software di base e software applicativo compreso nel prezzo di vendita: MS DOS 2.11, Tutorial, User interface, Diory, Sketch, Superwriter, Supercalc, Superplanner.

POINT 7

Il sistema Cluster Apricot Point 7 è una soluzione facile per l'automazione dell'ufficio, ad un costo ridotto. Si possono collegare fino a 7 micro-computers con possibilità di utilizzare contemporaneamente i programmi e gli archivi dell'unità centrale. Prezzo da lire 12.600.000.

VICTOR

È collegabile ai sistemi Apricot e il suo software è facilmente trasferibile. Prezzo da lire 6 900 000.

SISTEMI PC-COMPATIBILI

Possono essere collegati a P7 e P32 e sono gestiti dal Cluster controller come componenti della famiglia Apricot. Possono accedere ai dati del sistema o trasmetterli agli altri utenti.

APRICOT SI TROVA QUI.

• **ALESSANDRIA** - C I D COMPUTERS sas Via Tolstoi 17 - 15100 AL telefono 0131/42978 • **AOSTA** - INFORMATIQUE sas - A.V Conseil Commis 14 - 11100 AO - telefono 0165/362242 765173 • **BARI** - COMPUTER CORNER DI V CLEMENTE - Via Salvemini 46 A/B (M) - 70042 MOLA BARI - telefono 080/644361 • **BARI** - DEC SISTEMI DI DALENA CAPOBALE - Via Capruzzi 36/C - 70100 BA - telefono 080/580200 • **BERGAMO** - APA SOFT - Via Buonarroti 5/A - 24047 TREVIGLIO - telefono 0363/40597 • **BOLOGNA** - C R E MICROSISTEMI snc - Via Caravaggio 36 - 40033 CASALECCHIO R. - telefono 051/568020 • **BOLOGNA** - SORI srl P.zza Porta Mascarella 7 - 40126 BO - telefono 051/240602 17 - 230105 • **BOLZANO** - DATA PLAN - Via Cassa di Risparmio 9 - 39100 BZ - telefono 0471/47721 • **BRESCIA** - C.C.C DI TREVISI GIUSEPPE - Via S Bernardino 10 - 25032 CHIARI - telefono 030/712716 • **BRRSCIA** - IL COMPUTER - Via Benedetto Croce 11/13 - 25100 BS - telefono 030/42100 - 292308 • **CAGLIARI** - S.I I Via S Lucifero 95 - 09100 CA - telefono 070/663746 • **CATANIA** - C.S.E DI R CRISTINA - Via De Caro 54 - 95100 CT - telefono 095/402301 402370 • **CATANIA** - SISTEMI A PROCESSORI - Via G Matteotti 262 - 95045 MISTERBIANCO - telefono 095/301421 • **CATANIA** - T E A.M. - Via De Caro 35 - 95100 CT - telefono 095/402376 • **CREMONA** - PERSONAL E PROFESSIONAL COMPUTER - 26030 GADESCO P D - telefono 0372/81086 • **CUNEO** - E D P sas DI CAPRA R - Via Giacosa 1/A - 12051 ALBA telefono 0173/363951 • **FOGGIA** - SINFOR SDF - V.le Europa 44/46 - 71100 FG - telefono 0881/32579 70582 • **FIRENZE** - AMPEC - Via Panzani 13 - 50123 FI - telefono 055/219562 • **GENOVA** - ASAS - Via XX Settembre 2/13 - 16121 GE - telefono 010/581935 • **GENOVA** - MICRO PRO srl - Via Matteotti 50 (M) - 16011 ARENZANO - telefono 010/9123297 • **IMPERIA** - C.E.S.I srl - Via P Boselli 28 - 18011 ARMA TAGGIA - telefono 0184/43787 • **MILANO** - A.F L. - Via Bardelli 7 - 20131 MI - telefono 02/2366616 • **MILANO** - C.E COMMUNICATION ENGINEERING - P.zza Firenze 4 - 20154 MI - telefono 02/3182122 • **MILANO** - MICROFAST DI BAGNA C. Via Cuggiono 142 (F) - 20020 ARCONATE - telefono 02/9786592 • **MILANO** - PENTASISTEMI srl - Via Battistotti Sassi 31 - 20137 MI - telefono 02/741390-715340 • **MILANO** - PROMELIT - Via Volcava 15 - 20052 MONZA - telefono 039/734404 • **MILANO** - SELECON snc - Via F Soave 15 - 20135 MI - telefono 02/585202 5462791 • **MODRNA** - DATAGRAPH - Via G Pepe 13 - 41100 MO - telefono 059/335269 • **NAPOLI** - 3 G snc DI G GIAQUINTO - Via M D'Ayala 6 - 80125 NA - telefono 081/416697 • **NAPOLI** - GRAAL SYSTEMS - Via Petrarca 15 - 80100 NA - telefono 081/7695827 • **NAPOLI** - LIGUORI EDITORE - Via Mezzocannone 19 - 80134 NA telefono 081/203606 • **PADOVA** - SIC ITALIA - Via Palizzo 23 (F) 35100 PD - telefono 049/776588 • **PESCARA** - BINA TRADING sas Via P Penna 71 - 65100 PE - telefono 085/414055 • **PERUGIA** - SIGE DI BENUCCI - Via Bonlempi 21/25 - 06100 PG - telefono 075/393434 • **PISA** - IT LAB - Via XXIV Maggio 101 - 56100 PI - telefono 050/501359 • **PAVIA** - BYTE ELETTRONICA - P.zza Carducci 11 - 27058 VOGHERA - telefono 0383/212280 • **PAVIA** - M3 COMPUTERS - V.le C. Battisti 4/A - 27100 PV - telefono 0382/31087 • **RAVENNA** - ELCOS - Via Naviglio 11 - 48018 FAENZA - telefono 0546/28387 • **RAVENNA** - S.H.R. - Via Faentina 175/A - 48010 F ZARATTINI - telefono 0544/463200 • **ROMA** - EASY BYTE - Via G Villani 24/26 - 00179 ROMA - telefono 06/7811519 • **SALERNO** - GRAAL SYSTEMS - Via P Grisignano 4 - 84100 SA - telefono 089/321781 • **TORINO** - CI ESSE DI CASTAGNA PAOLO - Via Arona 8 - 10145 TO - telefono 011/774388 • **TORINO** - M.S.E - C.so Regio Parco 42 - 10153 TO - telefono 011/238766 • **TORINO** - MISTER PERSONAL DI VAUDAGNA - C.so Matteotti 13 - 10083 FAVRIA C.SE - telefono 0124/428344 • **TORINO** - ZUCCA COMPUTER sas - Via Tripoli 179 - 10137 TO - telefono 011/352262 • **TRIESTE** - RLMA COMPUTERS & C - Via Laghi 4/E - 34100 TS - telefono 040/793211 • **TRAPANI** - CARDILLO COMPUTER SYSTEM - Via Orti 63 - 91100 TP - telefono 0923/27981 • **VARRSE** - N T srl NEW TECHNOLOGIES - Via Serio 4 - 21057 OLGIATE OLONA - telefono 0331/640017 • **VICENZA** - TODARO LUCIANO - Via J Da Ponte 51 - 36061 BASSANO GRAPPA - telefono 0424/22810

Milanofiori - 20089 Rozzano - Strada 7 - Palazzo T3 - Tel. 02/8243741

GLIA MOLTO UNITA.

SPI Open Access

per PC IBM

di Corrado Giustozzi

Proseguendo nella nostra serie di prove di software per MS-DOS, ci occupiamo questo mese in dettaglio di Open Access, l'interessante pacchetto già presentato nel numero 37. Prodotto dall'americana SPI, Open Access è il principale rappresentante della moderna classe di applicazioni dette ambienti integrati sofisticati programmi che mettono a disposizione dell'utente tutta una serie di funzioni di calcolo, comunicazione, gestione di testi e di archivi, in una medesima struttura. Un ambiente integrato, cioè, implementa in un unico substrato tutte quelle risorse informatiche di supporto alle usuali necessità di office automation permettendo di usare un solo computer ed un solo programma per risolvere i tipici problemi di segreteria o di gestione di piccole aziende. In un ambiente integrato ogni funzione è a portata di mano, per passare i dati dello Spreadsheet al Word Processor, non occorre salvarli, uscire dallo Spreadsheet, caricare il Word Processor, convertire il file intermedio in formato compatibile con quello del Word Processor ed infine leggerlo. Basta la pressione di pochi tasti per trovarsi rapidamente nell'ambiente giusto, con i dati già disponibili nel formato corretto. Oltre a ciò un ambiente integrato permette lo scambio di dati con altri host, e la rappresentazione grafica dei dati sotto forma di diagrammi commerciali a barra, a torta e così via. In questo modo si realizza quella integrazione tra dati, informazioni e procedure, tipica delle applicazioni su mainframe, e che costituisce la vera risorsa messaci a disposizione dai computer. Avere dati e informazioni frammentari e spezzettati è inutile, l'integrazione del patrimonio informativo di un'azienda permette invece una completa gestione delle informazioni, per avere sempre il dato giusto, nel formato corretto ed al momento opportuno. Open Access in particolare comprende un sofisticato Data Base Management System, uno Spreadsheet, un generatore di grafici commerciali, un Word Processor, un Communication Link ed un inedito Time Manager, ossia un'agenda-calendario con gestore di appuntamenti. La sua struttura di base segue da vicino i moderni standard qualitativi per il disegno ergonomico del software uso dei colori per evidenziare i messaggi sullo schermo, impiego costante dei tasti funzione del PC, semantica consistente dei comandi nei vari moduli, largo uso di finestre e menu "pop-up", funzione di help in linea. Insomma, sulla carta si tratta di un prodotto piuttosto valido, fra l'altro la versione in prova questo mese non è la stessa cui si riferiva la panoramica del n. 37 (v. 1.00) ma una release successiva (v. 1.01), come accennato in quell articolo. In questa nuova versione sono stati aggiunti alcuni comandi ed è stata aumentata la velocità di esecuzione ottenendo un prodotto ancora più valido del precedente. Per questo motivo il distributore italiano ha deciso di non mettere affatto in commercio la versione 1.00, ed infatti da questo mese comincerà la distribuzione ufficiale della 1.01, questa sarà tra l altro completamente tradotta in italiano. La versione nazionale non ci è giunta in tempo per questa prova, e quindi non ve ne possiamo riferire, ma ci è stato assicurato che sarà tradotta e verificata da italiani, per evitare gli svarioni linguistici della precedente versione. E con questo passiamo ad incominciare.

Requisiti hardware e generalità

Iniziamo il discorso inquadrando la struttura di base di Open Access ed i requisiti hardware del PC che lo dovrà ospitare.

Le scelte di fondo operate dai progettisti di Open Acess sono andate in direzione diversa dalle altre riscontrabili in prodotti similari quali Symphony (provato sullo scorso numero). Mentre Symphony risiede per intero in memoria centrale, privilegia le

operazioni in RAM ed usa il disco assai poco e solo per lasciarvi i file permanenti, Open Access è spezzettato in diversi overlay che risiedono su disco e vengono caricati in memoria solo quando servono, e tende ad un massiccio utilizzo della memoria di massa per depositarvi file di lavoro, workaree temporanee e dati intermedi. Le due filosofie hanno vantaggi e svantaggi, e nessuna è definitivamente migliore dell'altra. Symphony è più veloce, ed in ogni momento si può cambiare ambiente avendo sempre tutto in linea, di contro obbliga ad avere almeno 512 Kbyte di RAM se si vuole una workarea di dimensioni ragionevoli. Open Access richiede solo 192 Kbyte di RAM, ma i frequenti accessi ai dischi per swappare i dati e/o i segmenti di programma ne rendono pesante l'uso con i floppy, consigliando caldamente l'acquisto del winchester. In caso contrario nel drive A devono risiedere i moduli di programma ed in quello B i dati.

Per quanto riguarda le altre necessità hardware, le capacità grafiche di Open Access richiedono la presenza di una scheda grafica e di un monitor a colori, che possono anche essere quelli originali IBM in quanto non vi è la necessità di mostrare i grafici assieme agli schermi di testo. La versione 1.01 supporta anche il dump su stampanti grafiche e l'uscita su plotter, che avviene tramite RS-232 in formato HP-7470.

Open Access viene fornito su sei dischetti, di cui tre contengono il programma vero e proprio. Sul primo si trova il modulo di boot e quello contenente le routine di utilità (configurazione ecc.), il secondo contiene i moduli Communication, Word Processor, Time Manager e l'Editor per le Macro, il terzo i moduli Information Management, Spreadsheet e Graphics. Sul quarto e quinto dischetto sono riportati dei file di lavoro a scopo di dimostrazione ed istruzione all'uso, mentre sul sesto è contenuta una piccola libreria di driver per la configurazione hardware, da utilizzare in fase di installazione del programma. La confezione è costituita da un grosso raccoglitore ad anelli in cartone che ospita il voluminoso manuale, una guida rapida e la mascherina in plastica per i tasti funzione del PC. Tutto sommato la presentazione è piuttosto sobria, per non dire povera, per un prodotto di questa classe e di questo costo si poteva forse fare qualcosa di più. Il manuale è praticamente un enorme tutorial, formato da sette sezioni, una per ambiente più un'introduzione, ed una appendice che funge anche da reference manual.

Produttore:
Software Products International
USA
Distributore per l'Italia:
SVPT
Via Val Cristallina, 3
00141 Roma
Prezzo:
L. 1.900.000 + IVA

Concetti di base

Open Access si basa su di un concetto di base completamente diverso da quello, ad esempio, di Symphony. In Symphony, ricordiamo, l'ambiente di lavoro principale è lo Spreadsheet, ed ogni altro ambiente è solo un modo diverso di vedere i dati riga/colonna del foglio, ciò permette di agire su più ambienti allo stesso momento, aprendo più finestre di tipo diverso sulle opportune aree del foglio di lavoro. In pratica Symphony dispone di un solo ambiente, personalizzabile in più modi. In Open Access invece la distinzione tra i vari ambienti è totale, non esiste un substrato comune tra di essi, e non si può avere contemporaneità di azione su diversi ambienti, ma si può agire solo su uno alla volta. In effetti gli ambienti vengono implementati con moduli di programma separati, che a volte non risiedono neppure sullo stesso dischetto, e quindi per passare dall'uno all'altro Open Access deve effettivamente cancellarlo dalla memoria e caricarne un altro, mantenendo in qualche modo una copia dei dati che devono essere visti dal modulo di arrivo. La comunicazione tra i vari ambienti avviene quindi in modo batch, per mezzo di file intermedi: all'uscita da un modulo viene automaticamente creato un file scratch contenente i dati da passare; quindi viene lanciato il modulo richiesto, il quale sempre automaticamente va a leggere il file e quindi lo cancella. Questo modo di operare ha qualche vantaggio: innanzitutto essendo l'utente obbligato, all'uscita di una sessione in un ambiente, a dichiarare al programma se vuole terminare il lavoro o se vuole entrare in un ambiente diverso trasferendovi alcuni dati, il programma è in grado di proporre i soli passaggi "ragionevoli" (ad esempio non ha senso mandare un elenco di appuntamenti allo Spreadsheet); in secondo luogo, ogni modulo può liberamente gestirsi i propri file nel formato più adatto all'applicazione senza doversi preoccupare della necessità di condividere lo stesso file con gli altri moduli, sapendo che quando necessario invierà al richiedente una copia dei dati formattata in modo opportuno; infine, essendo il verso del passaggio noto al momento in cui viene effettuato, il programma è in grado di creare il file intermedio nel formato adatto al modulo ricevente. L'utente può comunque salvare i propri dati in un formato standard comprensibile da tutti i moduli, denominato SIF (Standard Interchange Format). Ciò è stato fatto per permettere la lettura di file redatti in formati esterni. Un'apposita utility di Open Access permette infatti di leggere file in

Lo schermo di apertura di Open Access. Come si vede è stata aperta la finestra di help.

La confezione non particolarmente lussuosa

formato DBase II (.DBF), Data Interchange (.DIF) e ASCII (.TXT), convertendoli in formato .SIF. Senza questo standard interno di formato avrebbero dovuto essere implementate tante conversioni quanti sono i formati utilizzati dai vari moduli, una soluzione senza dubbio poco efficiente.

L'ambiente principale di Open Access è comunque il Data Base, o modulo di Information Management come viene definito nel programma. Basato su di una struttura relazionale, il Data Base permette una gestione veramente flessibile delle informazioni che poi possono essere inviate agli altri ambienti.

La struttura semantica di Open Access è ben stabilita, essendo obbligati a lavorare su di un ambiente per volta, l'uso delle finestre è limitato alla presentazione di comandi, menu di opzioni o help. Le finestre sono quindi gestite dal programma, che le apre quando servono e le chiude quando non servono più. In questo modo lo schermo non viene ingombrato inutilmente da lunghe liste di opzioni, in quanto queste vengono presentate solo quando è effettivamente necessario operare una scelta. I tasti cursore del PC mantengono il loro ovvio significato di spostamento, anche assieme ai tasti End o Page, a seconda dell'ambiente, però, lo spostamento può essere interpretato in modo diverso: di una riga, di un paragrafo, di una pagina, di un record, di un campo. I tasti funzione sono divisi in due classi: dipendenti dal contesto e no. Per questi ultimi, che non mutano il proprio significato nei vari ambienti, è disponibile una mascherina mnemonica da applicare sulla tastiera del PC. Le loro funzioni vanno dall'apertura di una finestra di menu (stavolta dipendente dal contesto) alla richiesta di help. Usatissimi in tutti gli ambienti sono i tasti F10 e ESC, denominati rispettivamente Do (fai) e Undo (torna indietro). Il primo ha sempre il significato di "seleziona" o risponde affermativamente alle domande, il secondo significa "lascia perdere, cancella", o risponde no. Particolarmente utile è poi il tasto Search, premendolo in risposta ad una richiesta di un nome di file, questo tasto fa aprire una finestra in cui vengono mostrati tutti i file disponibili appartenenti al tipo opportuno, per cui la scelta può essere fatta semplicemente muovendosi col cursore sul file giusto e premendo Do. Discussa quindi la struttura del programma in generale, passiamo a vedere più da vicino i vari moduli che lo compongono.

Il modulo Information Management

Il modulo di Information Management è, dicevamo, l'ambiente principale di Open Access. In effetti ci troviamo davanti ad un potente programma di Data Base, basato su di una struttura relazionale e Form-oriented. Le informazioni da archiviare possono essere distribuite su più di un file di formato diverso, e ricostruite dal programma per mezzo di semplici comandi. Ciò è quello che si fa normalmente sui mainframe, ma è difficile da trovare sui personal. Come esempio di questa struttura si può pensare ad un file contenente gli ordini ricevuti da una fabbrica durante il mese; ogni record conterrà come dati identificativi dell'ordine un numero progressivo, il codice del prodotto ordinato, la quantità ordinata, la data dell'ordine ed il codice del cliente. Non c'è bisogno di riportare per esteso in ogni record la descrizione dell'articolo e la ragione sociale del cliente, questi dati possono essere memorizzati una volta per tutte in due file di consultazione, ed agganciati quando serve per mezzo di appositi campi chiave (i codici di prodotto e di cliente, appunto). Per effettuare interrogazioni sugli archivi l'utente ha a disposizione semplici comandi ispirati ad un linguaggio Query sviluppato dall'IBM qualche anno fa, denominato SQL. Con esso si possono fare estrazioni, ordinamenti, agganciamenti fra file diversi eccetera, ottenendo come risultato un file

In ambiente Data Base il programma ci chiede il nome del file da leggere. Se in risposta premiamo il tasto Search appare l'elenco dei file disponibili.

Se ora premiamo Help due volte si aprono la finestra di aiuto e quella che ci ricorda l'uso dei tasti.

di lavoro costituito dai record soddisfacenti i criteri di selezione, e costituiti dai soli campi desiderati. Questo file può essere salvato, stampato, sortato, interrogato per campi chiave, aggiornato e chi più ne ha più ne metta, non escluso l'invio ad un altro modulo

È possibile inoltre definire le proprie maschere per l'ingresso dei dati, in fase di definizione dei campi se ne possono stabilire gli attributi, così che il programma è in grado di controllare i dati sin dal loro ingresso Ad esempio si posso definire campi solo numerici, solo interi, di tipo vero/falso o in formato data Per alcuni di essi fra l'altro il programma adotta su disco una rappresentazione compatta, per risparmiare spazio. i campi data e gli interi occupano così solo due byte.

L'immancabile mascherina in plastica riportante l'uso dei tasti definibili

Il modulo Spreadsheet

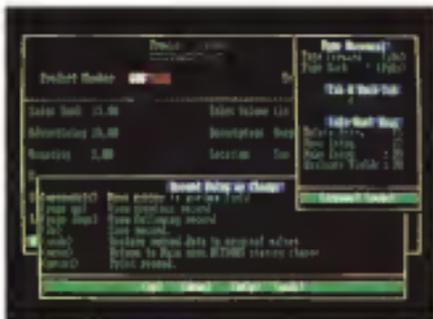
L'ambiente Spreadsheet è secondo per importanza al precedente. In esso si possono effettuare calcoli fra valori sia impostati ex novo che ricevuti dal Data Base La sua estensione è di 3.000 righe per 216 colonne, ossia ben 648.000 celle! Cosa molto importante, questa ampiezza è indipendente dall'estensione della memoria centrale, e quindi potenzialmente è tutta a disposizione dell'utente Ciò in quanto questo modulo sfrutta la tecnica cosiddetta di memoria virtuale, utilizzando il disco come "appoggio" per allocare temporaneamente aree del foglio attualmente non in uso, mentre in RAM rimane la zona effettivamente usata

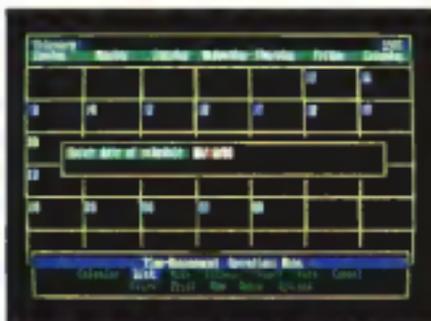
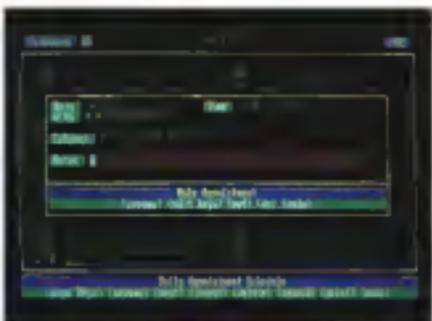
Il foglio può venire splittato in sei finestre indipendenti, e dispone di un nutrito set di funzioni matematiche (trigonometria, logaritmi, media varianza e deviazione standard) e finanziarie. Ma la sua caratteristica forse più interessante è la funzione di Goal Seeking, ossia ricerca all'indietro. Ogni Spreadsheet permette il cosiddetto calcolo "what-if", consistente nel cambiare qualche valore (variabile indipendente) e vedere come cambiano di conseguenza le variabili che da esso dipendono. Il Goal Seeking permette invece di agire al contrario. definendo il valore che si desidera avere come variabile dipendente, il programma risale al valore che deve avere la variabile indipendente perché la richiesta dell'utente sia verificata entro una data approssimazione

Il modulo Graphics

In questo modulo, dobbiamo dire, si è sbizzarrita la fantasia dei progettisti Questo generatore di grafici è uno dei più completi e versatili che ci sia capitato di vedere. La sua caratteristica principale è quella di poter rappresentare funzioni di due variabili sotto forma di istogrammi tridimensionali I controlli a disposizione sono innumerevoli, e permettono di aggiustare l'apparenza estetica del grafico soddisfacendo

Un esempio di uscita grafica su plotter Per la complessità del disegno che implica il calcolo di molte linee nascoste il 7475 usato ci ha messo oltre quattro minuti per completare l'opera

Abbiamo finalmente trovato il record cercato La maschera è registrata su un file opposito

Ed anche qui abbiamo un Help su ciò che possiamo fare

Due immagini dello Spreadsheet. Come si vede il campo è stato splittato in due finestre. A destra ancora un esempio di Help

Il calendario del modulo Time Manager. A sinistra la fase in cui si chiede di inserire un appuntamento. A destra l'effettivo ingresso dei dati dell'appuntamento

anche i palati più esigenti. Si possono definire etichette, tipo di campitura, modalità di rappresentazione, perfino l'angolo dell'assonometria del disegno tridimensionale. Nei diagrammi a torta si possono esplodere quanti segmenti si vuole, è possibile sovrapporre più tipi di rappresentazioni e perfino memorizzare una sequenza di immagini già realizzate per ottenere una "proiezione" in successione. I disegni possono essere trasferiti su plotter o stampante grafica, nel primo caso mantenendo i colori.

La velocità di disegno sullo schermo è entusiasmante, molto maggiore di quella della versione precedente, in questo modo è possibile "rifinire" un grafico in pochi minuti per mezzo di tentativi successivi, realizzando un prodotto decisamente di prim'ordine.

Il modulo Word Processor

L'ambiente Word Processor è piuttosto usuale, sono disponibili i normali comandi di movimento, inserimento e cancellazione, ricerca e sostituzione, spostamento di brani anche da/verso file esterni, giustificazione e controllo degli attributi di stampa. È possibile realizzare testi personalizzabili in stampa con valori letti da un file, ed includere nei testi parti dello Spreadsheet.

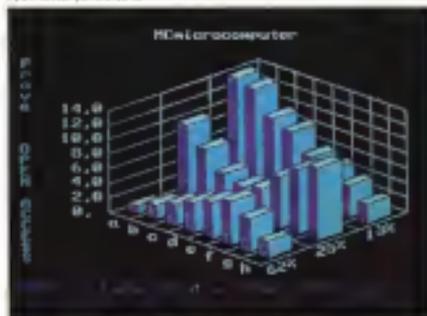
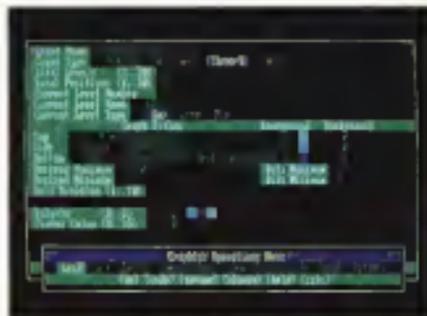
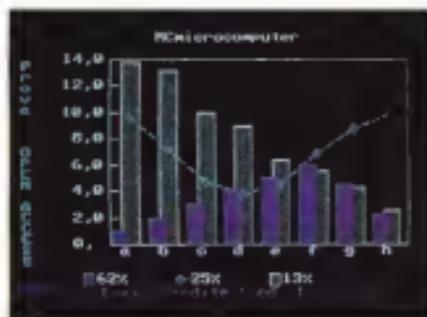
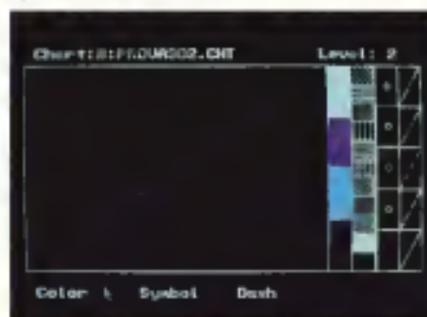
Il modulo Communication

Da questo ambiente si possono gestire le comunicazioni via modem fra il proprio PC ed altri computer o banche dati. È possibile definire una sequenza di autodial, modem permettendo, e configurare i parametri della trasmissione e del modem. I file ricevuti vengono memorizzati su dischetto, da dove possono essere ripescati al momento opportuno.

Il modulo Time Manager

Questa parte di Open Access, decisamente poco comune, svolge funzioni di agenda e calendario, permettendo la registrazione di appuntamenti ed impegni. È possibile impostare, cancellare, modificare e visualizzare gli appuntamenti memorizzati, mantenere il controllo sulle fasce orarie evitando impegni in determinate ore, programmare impegni fissi e così via. Si tratta di una funzione piuttosto slegata dalle altre del pacchetto, ed inserita evidentemente per completezza nell'ottica della totale automazione delle funzioni d'ufficio cui mira Open Access.

Utilizzazione

Per quanto riguarda l'utilizzazione di Open Access bisogna premettere una cosa: il programma è talmente complesso e voluminoso che prima di riuscire a padroneggiarlo pienamente deve passare un bel po' di tempo. Comunque è piuttosto semplice imparare ad usarlo ad un buon livello, e lo si fa piuttosto in fretta. Crediamo comunque che non si possa dare indiscriminatamente in mano ad un manager (categoria notoriamente refrattaria alle cose informatiche) o, peggio, ad una segretaria, senza qualche ora spesa in un'acconcia istruzione del soggetto. A questo proposito vogliamo notare che il distributore italiano ha in progetto la realizzazione di brevi corsi d'istruzione su Open Access per rivenditori

Una bella immagine tridimensionale: la velocità di tracciamento è molto elevata

La schermata di configurazione del grafico di sinistra. Come si vede le opzioni disponibili sono moltissime

Una sovrapposizione di grafici di tipo diverso

Le opzioni di tracciamento dei grafici a due dimensioni. Si possono scegliere colore e tipo di linea, schema di campitura e così via

ed utenti, un'iniziativa interessante, che dovrebbe riscuotere un certo successo se, come sembra, il pacchetto prenderà piede anche sul nostro mercato.

Tornando alle note d'uso, ci è sembrato che il programma si lasci usare volentieri. La sua filosofia piuttosto "tradizionale" permette sane gestioni dei propri dati, ammesso che si sappia con precisione cosa si voglia ottenere da essi. Ciò che può facilmente capitare è che l'utente si perda nella congerie di ambienti, comandi e strutture, perdendo di vista le cose necessarie e disperdendo tempo ed energia nel "giocare" con le informazioni. Una rigida pianificazione è invece necessaria se si vogliono fare le cose per bene.

Ribadiamo che Open Access senza il winchester è sprecato; fra l'altro la notevole messe di file e filetti che ogni modulo tira fuori da tutte le parti rischia di congestionare ben presto i 320 K del povero floppy, costringendo le testine a viaggiare su e giù per recuperare i pezzetti di file sparsi qua e là. Meglio buttare tutto sull'hard disk, oltre a non dover cercare il modulo di programma sui vari dischetti, non c'è più preoccupazione di riempire troppo presto 10 Mega!

Notiamo che i dischetti di Open Access sono protetti nel solito modo che consente la copia, ma non il boot. Sia che lo si faccia da floppy che da winchester, all'atto del lancio iniziale il programma vuole trovare il dischetto originale nel drive A:, altrimenti si rifiuta categoricamente di partire.

Conclusioni

Vogliamo finalmente trarre qualche conclusione su Open Access? Attenzione, però, viene spontaneo alla mente il paragone con Symphony, che però a parer nostro non è totalmente proponibile. Open Access è un prodotto "serio", "posato", "tradizionale" anche se innovativo, mentre Symphony è un oggetto più snello, e meno rigido, piuttosto anticonformista e per certi versi più versatile anche se meno potente (ad esempio come Data Base). Per cui, onde evitare di travisare il tutto, giudichiamo Open Access per sé. Ci sembra innanzitutto che si tratti di un prodotto ben pensato e ben realizzato, utile ed utilizzabile non solo sulla carta, ma sul campo. La sua stessa struttura ne fa uno strumento di lavoro serio, e non un gadget. La sua complessità d'uso, però, ne limita l'uso completo a chi sappia cosa sta facendo, e sappia dove vuole arrivare. È vero che Open Access è programmabile, ma per farlo serve un "programmatore di Open Access", il che non sposta il problema. Il prezzo è effettivamente ragionevole se confrontato alle prestazioni. Certo non è poco, ma se andate a comprare un Data Base, uno Spreadsheet ed un Word Processor (supponendo che non vi serva un Communication link) probabilmente pagherete una cifra non molto inferiore al milione e mezzo: ed in più qui avete un poderoso generatore di grafici ed un simpatico Time Manager, che potete anche non usare ma, crediamo, finirete con l'apprezzare. Fra l'altro l'Open Access che sarà distribuito parlerà italiano, ed anche questo sarà probabilmente un grosso vantaggio per l'inserimento in azienda. E quindi? Beh, se avete un IBM col winchester e cercate una soluzione ai problemi del vostro ufficio, fate un pensierino anche su Open Access!

Nella prova del QL pubblicata sul numero di luglio/agosto dello scorso anno il poco tempo e spazio a disposizione ci avevano costretti ad accennare solamente di sfuggita ai quattro programmi applicativi forniti in dotazione al computer.

Ora a poco più di dodici mesi dalla presentazione ufficiale avvenuta a Londra, il QL è finalmente arrivato anche in Italia e l'occasione ci sembra opportuna per riprendere l'argomento ed esaminare da vicino questa ormai famosa suite di programmi.

Per chi ancora non lo sapesse, si tratta di un word processor (Quill) un data base (Archive) un tabellone elettronico (Abacus), ed un programma di business graphics (Easel).

Prima di descriverne le caratteristiche vogliamo però parlare brevemente di quella che è la situazione attuale del QL, con particolare riferimento alla sua libreria software.

Abacus, Archive, Easel, Quill:

i quattro programmi del Sinclair QL

di Maurizio Bergami

Il QL un anno dopo

Con il QL la Sinclair ha seguito ancora una volta la sua sorprendente politica di vendita per corrispondenza, con tempi di consegna di due/tre mesi contro i soliti 28 giorni dichiarati nella pubblicità. La cosa certamente non ha contribuito a dare al QL una buona immagine e, quello che è peggio, un lancio prematuro ha messo in mano ai primi utenti un prodotto chiaramente incompleto e pieno di bug.

Una situazione del genere si era già parzialmente verificata con lo Spectrum, questa volta però la Sinclair non ha fatto bene i suoi calcoli ed il magico nome di Sir Clive non è bastato a far vendere il QL, il numero di unità consegnate nell'84 è stato inferiore a 45.000, una cifra che lo stesso Sinclair ha definito "disappointing".

Contemporaneamente il software si è fatto attendere a lungo e la previsione, fatta da Nigel Searle, Managing Director della Sinclair, di "50 programmi disponibili per il QL entro la fine dell'84" si è rivelata largamente errata per eccesso.

Anche se con un po' di ritardo, le software house inglesi sembrano tuttavia essere finalmente entrate in confidenza con il QL, ed il numero dei programmi ad esso dedicati sta aumentando con rapidità. Ciò è dovuto, in parte, anche alla brusca riduzione di prezzo delle microcartucce, passate da 5 a 2 sterline (la notizia farà sicuramente felici anche i possessori dello Spectrum).

Il software attualmente in circolazione è composto per la maggior parte da linguaggi programmativi, tra i quali troviamo Pascal, Forth, Lisp, BCPL (un predecessore del più noto C), APL e ben due assembler.

Molto interessante è un toolkit commercializzato dalla stessa Sinclair che, tra le tante possibilità, permette di controllare il

multi-tasking da Basic con estrema facilità e dota il Super Basic di un editor di schermo (finalmente!)

Fino ad ora di giochi non se ne sono visti, a parte lo strepitoso programma di scacchi della Psion, che mostra la scacchiera in tre dimensioni, ed un paio di banali avventure. Peccato, perché la velocità del 68008, in unione con l'elevata risoluzione dello schermo (soprattutto per quanto riguarda il colore) e la quantità di memoria, permetterebbe di ottenere risultati eccellenti.

Cosa singolare, i programmi citati sono praticamente tutti di software house in precedenza sconosciute, i grandi nomi si sono tenuti fino ad ora fuori della mischia e stanno probabilmente aspettando una più vasta diffusione del QL.

Prima ancora del software, comunque, sono apparsi gli accessori hardware, almeno tre ditte (Quest, CST e Medic Datasystems) hanno già da alcuni mesi in catalogo un'interfaccia per floppy disk e numerose espansioni Ram, da 64 a 512 Kbyte.

Insomma, anche se piano, le acque si stanno muovendo e nell'85 il QL avrà probabilmente il successo di vendite che ha sempre arriso a tutti i prodotti Sinclair, accompagnato dalla consueta invasione di libri, programmi, interfacce ed accessori di tutti i tipi e per tutti i gusti.

I programmi applicativi

Veniamo ora ai quattro programmi applicativi. Sviluppati dalla Psion per conto della Sinclair, i programmi forniti assieme al QL sono in grado di soddisfare le esigenze di base di una vastissima fascia di utenti.

In fase di progetto la Psion ha posto l'accento principalmente sulla facilità d'uso, rendendosi conto che la maggioranza degli acquirenti del QL sarebbe stata composta da persone alla prima esperienza con programmi di questo tipo.

Tutti e quattro i programmi conservano una struttura comune, a partire dal formato di schermo.

In alto sul display si trova sempre la cosiddetta area di controllo, che contiene una serie di prompt (dipendenti dalle opzioni disponibili in quel momento) ed il significato dei tasti funzione. In basso vi sono invece le due linee di status, che permettono di avere sotto controllo lo stato corrente del programma (fa eccezione Archive, in cui queste linee sono sostituite da una finestra usata per impartire i comandi al data base).

Con il tasto funzione F1 si accede alle numerose pagine di help, conservate su microdrive, che sono collegate fra loro da una struttura ad albero: da ogni pagina è possibile richiamare una o più pagine successive per avere maggiori dettagli sull'argomento voluto, sino all'esaurimento delle informazioni disponibili.

In questo modo si ha a disposizione un manuale "intelligente" direttamente sullo schermo, tanto efficace che in molti casi si può fare benissimo a meno del manuale su carta.

Nonostante il sistema operativo del QDOS sia multitasking, è possibile far girare soltanto un programma alla volta, questo fatto ha procurato alla Sinclair moltre critiche, ma pochi si sono resi conto che il multitasking dei programmi applicativi è impedito in partenza dalla poca Ram del computer: nei circa 90 Kbyte di Ram a disposizione (gli altri sono dedicati al video) non può entrare che un unico programma. In compenso è possibile trasferire un file da un programma all'altro, dopo averlo registrato nell'apposito formato di scambio.

La possibilità di collegare il QL ad un monitor o ad un TV ha convinto la Psion a prevedere tre differenti formati per l'output su video: 40, 64 o 85 colonne. I valori di default sono rispettivamente 64 colonne per il TV e 80 per il monitor, il modo a 40 colonne usa caratteri estremamente leggibili per la loro grandezza, ed è consigliabile a chi adopera un televisore di qualità non troppo elevata che entrerebbe in crisi con le 64 colonne.

Un problema di tutti e quattro i programmi è quello della lentezza: ciò è dovuto in parte al software, ma soprattutto all'uso dei microdrive come memoria di massa. L'utilizzo dei floppy disk sopra menzionati permette di ottenere notevoli miglioramenti, come abbiamo potuto constatare di persona all'ultima ZX microfair, ma è un rimedio indubbiamente costoso.

Rimane il fatto che, anche con i microdrive, la velocità rimane sufficiente per un uso non a livello professionale del computer, e cogliamo l'occasione per ribadire una volta di più la nostra convinzione che il QL sia una macchina splendida per l'hobbista, ma decisamente sconsigliabile al professionista a causa della sua relativamente scarsa affidabilità.

Passiamo ora a descrivere i quattro programmi uno per uno, le note fornite sono volutamente ridotte all'essenziale, per evitare di fare un semplice sunto del manuale, ma sono in compenso integrate da numerose

Un monitor per il QL

Le foto di questo articolo sono state fatte con il monitor QL 14 della ditta inglese Prism, gentilmente messoci a disposizione dalla Disitaco.

IL QL 14 è stato costruito appositamente per il QL, ciò non vuol dire, come potrebbero pensare i maligni, che essendo nero si intona bene al computer, ma che è capace di visualizzare tutte le 85 colonne dello schermo del QL.

Non tutti i monitor infatti ce la fanno, ma questo non significa che siano difettosi: la colpa infatti è del QL, che, tanto per cambiare, genera un segnale video non perfettamente standard.

Per tornare al QL 14 va osservato che si tratta di un monitor dichiaratamente economico, con una banda passante di soli 12 MHz, che tuttavia è in grado di offrire risultati eccellenti se paragonato ad un normale TV.

L'immagine fornita è brillante (forse persino troppo) e ben contrastata; il monitor e anche dotato di uno schermo scuro antiriflesso, eventualmente rimovibile con poca fatica.

Chi è abituato all'uso del TV nota subito la notevole chiarezza dell'immagine e, soprattutto, la scomparsa di ogni forma di tremolio dei caratteri. L'unico controllo disponibile è quello di contrasto, situato frontalmente dietro lo sportellino con il logo Prism, sempre sul frontale si trova l'interruttore di accensione ed un led rosso che segnala lo stato di on.

roso materiale fotografico, ampiamente commentato

Quill

Incominciamo da Quill, il programma di trattamento di testi. La sua principale caratteristica è di essere basato sulla filosofia del What You See Is What You Get. Questo significa principalmente che i caratteri che compongono il testo vengono mostrati sul video proprio come verranno stampati sulla carta, ad esempio, una frase composta da caratteri sottolineati apparirà effettivamente sottolineata sullo schermo, e lo stesso discorso vale per gli altri tipi di caratteri, High Script e Low Script.

C'è anche il grassetto (Bold), che però viene semplicemente visualizzato con un diverso colore, per evitare di avere delle linee più lunghe di altre.

Le opzioni fondamentali di formattazione e di editing ci sono tutte, è possibile tra l'altro aggiungere titoli ad ogni pagina, in alto o in basso, spostare o cancellare blocchi di testo, inserire tabulazioni di quattro diversi tipi e molte altre cose ancora.

I nei del programma sono essenzialmente due: la lentezza (sia di Quill stesso che dei microdrive) e la tastiera non standard, entrambi sono ampiamente sopportabili da chi non è costretto a scrivere molto e soprattutto in fretta. Peccato però, perché per il resto regge benissimo il confronto con programmi ben più famosi (come il celebratissimo Wordstar) ed avrebbe potuto rendere il QL una perfetta stazione di word processing anche per giornalisti e scrittori.

Archive

Archive è un data base che può essere

Quill

Ecco un esempio di come appare il testo sullo schermo. Come si può vedere, i vari tipi di carattere vengono mostrati esattamente come verranno stampati sulla carta. Fa eccezione il grassetto, che è uguale al carattere normale ma di colore bianco anziché verde.
Nelle due linee inferiori viene indicato, assieme a numerose altre informazioni, il carattere (Typeface) attivo in quel momento.

La finestra in alto, chiaramente visibile nella schermata precedente, può essere eliminata con la pressione del tasto funzione F2. Normalmente essa contiene una sorta di mini-help relativo ai tasti funzione. Con la pressione di F1 si richiamano invece gli help veri e propri, composti da una serie di pagine conservate su microdrive. Da ognuna di esse è possibile accedere ad altre pagine con ulteriori informazioni sugli argomenti trattati nella precedente.

Quill permette di avere sullo schermo 80, 64 o 40 caratteri per linea. Il valore di default è 80 se si usa un monitor oppure 64 se è collegato un normale TV. Se il televisore non è di buona qualità anche 64 caratteri possono essere troppi e conviene passare a 40, ottenendo il risultato che si vede in questa schermata. La riga rimane comunque lunga 80 caratteri (se non si cambiano i margini) e lo schermo agisce come una finestra sulla pagina di testo, scrollando automaticamente.

Questa è la pagina di Design, che contiene tutte le opzioni disponibili per cambiare formato al testo.
Per accedervi bisogna premere prima F3 e poi la lettera d. Le varie possibilità vengono selezionate semplicemente premendo il tasto che corrisponde alla iniziale del loro nome.
Le scelte a disposizione sono molte, chiaro indice della versatilità di Quill.

usato a due diversi livelli di complessità. Al primo si comporta come una semplice raccolta di schede, il cui formato può essere definito a piacere. Su queste schede sono permessi ordinamenti, ricerche normali e condizionali e tutte le operazioni normalmente associate con un semplice archivio.

Al secondo livello Archive si trasforma in un potente database programmabile, dotato di un linguaggio di programmazione simile al SuperBasic. Programmare un database significa, in questo caso, definire delle procedure con il linguaggio a disposizione in modo da creare una versione "su misura" di Archive per ogni nuova applicazione. In altre parole, mentre nel primo caso, cambiando l'insieme di dati da archiviare, veniva modificato solo il formato del file, per adattarsi ai nuovi dati, nel secondo è tutto il programma che viene modificato, in modo da risultare efficace al massimo con quei dati.

Ovvio che il secondo livello consente una flessibilità non comune, ma altrettanto ovvio che ad essa corrisponde una complessità molto maggiore. Non per nulla Archive è l'unico dei quattro programmi che richiede un accurato studio del manuale per poter essere usato in maniera completa.

Abacus

Questo è il tabellone elettronico, composto da una griglia di 256 righe per 64 colonne; in pratica non sarà mai possibile riempirlo tutto, dato che la memoria a disposizione per dati e formule è di soli 15K, sufficienti per la maggioranza delle necessità domestiche (bilancio familiare e via dicendo), ma non per applicazioni complesse come il bilancio di una ditta, anche se molto piccola.

Archive

Archive è il più complicato dei quattro programmi in dotazione al QL, ma anche il più sofisticato. Sulla microcartuccia che lo contiene è registrato anche un file di esempio contenente i dati principali relativi agli stati di tutto il mondo. La schermata mostra i dati dell'Afganistan, che sono stati visualizzati con la successione dei tre comandi "look", "first" e "display". I record successivi possono essere richiamati con il comando "next".

Naturalmente non è necessario dover scorrere tutto un file per trovare il record cercato. Si possono effettuare ricerche usando il comando find seguito dall'informazione desiderata.
La foto mostra il record relativo all'Italia, ottenuto dando il comando find "Italy". Se ci fossero più record contenenti l'informazione Italy, Archive mostrerebbe il primo; quelli successivi potrebbero essere ricercati con il comando continue.

Oltre a find, un comando utilissimo è search, col quale si possono effettuare ricerche in base a determinate condizioni. Nell'esempio mostrato nella foto abbiamo fatto cercare ad Archive uno stato africano in cui sia parlata la lingua inglese. Dato che il nome dei campi che contengono continente e lingua sono rispettivamente continent$ e languages$ è bastato digitare search continent$ = "africa" AND languages$ = "english".

Ecco un altro esempio di ricerca tramite search.
In questo caso l'obiettivo era uno stato asiatico con popolazione maggiore di 200 milioni di abitanti.

Pur non raggiungendo le prestazioni di un Lotus 1-2-3, Abacus dispone di una potenza di calcolo decisamente elevata, come si rileva dalle numerose funzioni sia di testo che numeriche, e di una velocità soddisfacente.

La caratteristica più interessante di Abacus è sicuramente il sistema usato per l'etichettamento delle celle. Il testo inserito in una cella viene considerato automaticamente come un'etichetta, che può essere riferita ad un'intera riga o colonna. La cosa principale è che il programma, nella maggioranza dei casi, può capire da solo se l'etichetta si riferisce ad una riga o ad una colonna. Per esempio, basta mettere il testo "spese" all'inizio di una riga contenente una serie di numeri perché Abacus assuma automaticamente "spese" come etichetta di quella riga. Grazie alle etichette si possono scrivere le formule in modo molto più semplice che non con il tradizionale sistema di riferimento numero/lettera.

Estremamente utile è anche la possibilità di dividere lo schermo in due finestre, orizzontali o verticali, le quali possono essere posizionate su due parti diverse del tabellone, che altrimenti non entrerebbero contemporaneamente sullo schermo. Entrambe le finestre possono essere spostate a piacere e possono persino venire collegate fra loro, in modo che muovendone una anche l'altra si sposti automaticamente della stessa distanza.

Easel

Eccoci arrivati all'ultimo programma, Easel. Si tratta di un programma di business graphics, che consente di ottenere una rappresentazione grafica di uno o più insiemi di dati.

Abacus

Questo è un semplicissimo esempio di come può essere utilizzato Abacus. Il foglio che abbiamo costruito calcola le spese totali relative ad un periodo di quattro mesi; sullo schermo vengono mostrati anche i totali di ogni mese. Le schermate seguenti mostrano come è stato approntato il tabellone.

Per prima cosa è stata definita la parte "fissa" che comprende una serie di diciture. Non tutte sono state inserite casella per casella: ad esempio per i nomi dei mesi è bastato usare il comando row=month(col) ()+1). La scritta in alto è contenuta per intero nella casella C1, dal momento che la sua lunghezza supera il numero di caratteri disponibili sullo schermo, viene stampata anche sulle celle successive. Se queste fossero state occupate la cosa non sarebbe stata possibile.

A questo punto sono state inserite le formule, davvero molto semplici. È bastato infatti definire le celle da C15 a F15 come la somma delle celle sovrastanti della stessa colonna, il tutto grazie al comando sum. Con le formule al loro posto non è rimasto altro da fare se non inserire i numeri nelle varie caselle; in generale l'inserimento di un dato si ottiene posizionando il cursore sulla casella opportuna, digitando il numero e premendo Enter.

Il tabellone viene ricalcolato automaticamente ogni volta che viene variato un dato; questa caratteristica può essere disattivata per permettere l'inserimento di una serie di dati senza dover perdere tempo inutilmente. L'ultima riga del foglio mostra il totale relativo al quadrimestre: la cella C19 è stata definita come la somma di C15, D15, E15, F15. Per mostrare tutto il foglio sullo schermo è stato necessario disattivare i prompt con il tasto F2.

Una volta che i dati sono in memoria il grafico si ottiene con straordinaria facilità, semplicemente scegliendo da una serie di menu il formato (sono disponibili vari tipi di istogrammi, il diagramma a linea continua e quello a torta) e tutti gli altri elementi (assi, sfondo...). In ogni momento è possibile cambiare tipo di rappresentazione utilizzando lo stesso metodo.

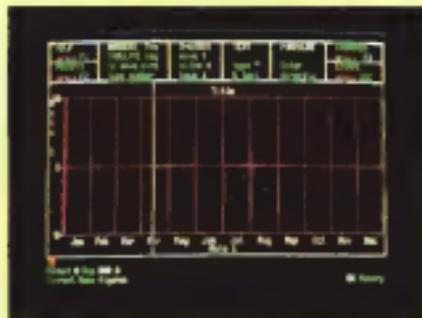
Un'interessante possibilità è quella che permette di definire delle formule per ottenere dei nuovi set di dati, con i relativi grafici, a partire da quelli già in memoria.

Easel è senza dubbio il più divertente e coreografico dei quattro programmi del QL, ma probabilmente anche quello di utilità più limitata per l'utente tipo di questo computer.

Conclusioni

Visto il costo del QL, i quattro programmi della Psion vanno considerati un po' come uno straordinario regalo. Non crediamo però che possa valere il discorso contrario, e cioè: l'acquirente in pratica compra i programmi e l'hardware arriva gratis, non lo crediamo perché questo equivarrebbe a proporre il QL come un computer professionale, mentre non lo è affatto. Il QL è invece, secondo noi, forse il più bel home computer attualmente sul mercato, ed è proprio in un uso domestico che i difetti di questi programmi in pratica scompaiono, lasciando visibili soltanto i numerosissimi pregi.

Anche la Sinclair se ne è finalmente resa conto, e nella sua nuova e massiccia campagna pubblicitaria, partita a febbraio, si rivolge non più al professionista, ma al "serious computer hobbyst", cioè all'appassionato di computer "serio", se si vuole, ma soprattutto appassionato.

Easel

Di tutti e quattro i programmi Easel è probabilmente il meno utile per l'hobbista in compenso è anche il più divertente da usare per le sue doti coreografiche. Con questo non vogliamo togliere nulla alle sue doti. Easel è sicuramente un eccellente package di business graphics comodo da usare ed estremamente versatile.

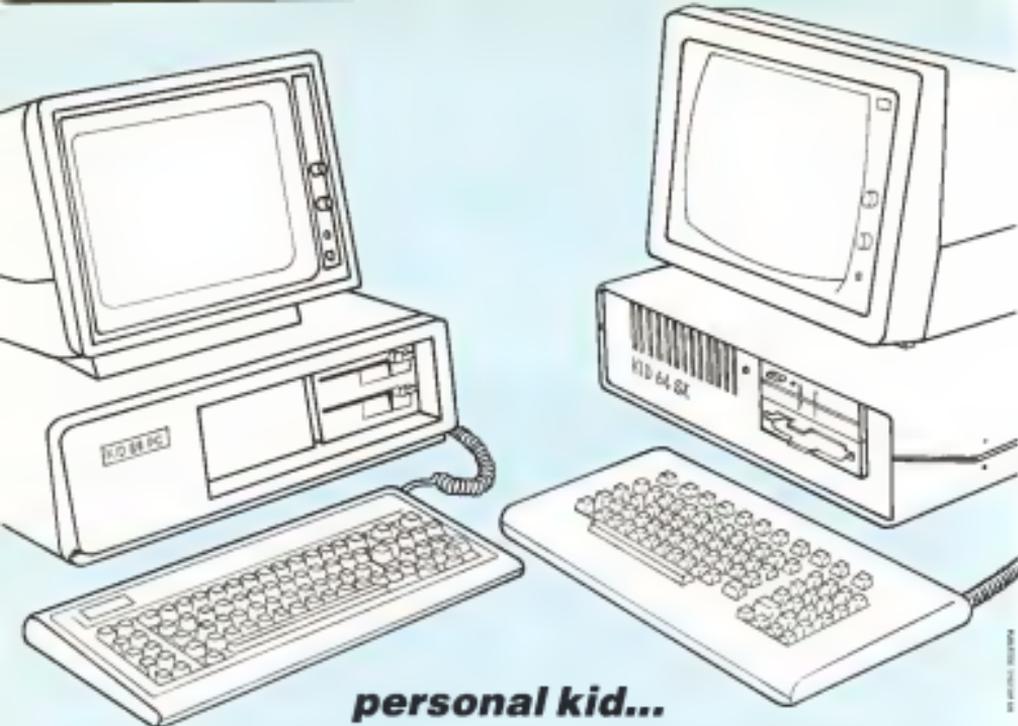
Tutti gli elementi che entrano a far parte della rappresentazione grafica possono essere modificati a piacere. La modifica può avvenire in due modi: selezionando un'altra delle numerosissime scelte offerte da Easel oppure ridefinendo l'elemento in questione con un mini-editor interno al programma. La schermata mostra i vari tipi di assi disponibili, lo spazio vuoto è riservato a quelli disegnati dall'utente. La selezione avviene posizionando la cornice bianca sul tipo di asse desiderato e premendo Enter.

I formati disponibili sono otto, l'ultimo è quello a torta che richiede un po' di tempo per essere disegnato.

La scelta di un formato non è definitiva, in ogni momento è possibile cambiare idea e passare ad un altro tipo di rappresentazione. Se nella stessa figura sono presenti più set di dati il programma inserisce automaticamente una finestra con la chiave del grafico.

Un interessante possibilità relativa ai diagrammi a torta è la messa in evidenza di un settore, che il programma ottiene staccando leggermente dalla torta lo spicchio interessato.

Le diciture vengono sempre posizionate automaticamente, tuttavia se gli spicchi sono troppo piccoli, c'è il rischio che alcune di esse si sovrappongano.

MC

personal kid...

più che compatibili

KID 64 SX l'unico Apple* compatibile dotato di Hard Disk da 10 MB e floppy da 640 MB integrati e software per il back up fisico e logico. Se il Kid 64 SX è troppo per te, puoi scegliere un Kid 6410 (con un Floppy da 143 K) o un Kid 6420 (con doppio floppy da 143 K) con la possibilità di effettuare l'up grading fino al Kid 64 SX entro il periodo di garanzia.
CPU 6502, coprocessore Z 80, 64 K RAM, uscita monitor a colori, uscita TV, compatibile DOS*, PRODOS*, CP/M*.

KID 88 PC compatibile IBM*, CPU 8088, coprocessore 8087 (opzionale), 128 K (256 K) RAM, 2 porte seriali RS 232 C, porta parallela Centronics, 9 slots di espansione, 2 floppy da 360 K, scheda grafica, compatibile MS-DOS*, CP/M86*.

*APPLE DOS, PRODOS, Trademark Apple Computers Corp.) *CP/M CP/M86, Trademark Digital Research) (IBM Trademark International Business Machinery) *MS-DOS Trademark Microsoft Corp.)

GARANZIA INTEGRALE DI 12 MESI

Via Di Vittorio 82 Tel (071) 8046305
60020 CANDIA ANCONA

GRADIREI RICEVERE INFORMAZIONI SU:

☐ KID 6410 ☐ KID 64 SX
☐ KID 6420 ☐ KID 88 PC

Nome
Indirizzo
Città Cap
Tel .. Professione

Currah µSpeech

per Spectrum

di Maurizio Bergami

Il Currah Microspeech è uno degli accessori più interessanti e divertenti tra i tanti disponibili per lo Spectrum.

Si tratta, come è facilmente intuibile dal nome, di un sintetizzatore di parola, che, grazie alla tecnica di sintesi utilizzata, permette di disporre di un vocabolario praticamente illimitato.

Le applicazioni possibili per un calcolatore parlante possono essere numerosissime, a patto però che la voce generata risulti effettivamente intelligibile senza difficoltà.

Vediamo allora se questo apparecchietto è davvero in grado di donare la parola a un altrimenti muto ZX.

Descrizione

Il Microspeech è alloggiato in un minuscolo contenitore di plastica nera, che su un lato presenta l'onnipresente connettore per il collegamento con lo Spectrum. Purtroppo manca la replica del connettore posteriore del computer, la scelta è stata sicuramente imposta dalle ridottissime dimensioni dell'unità ed è in parte giustificata dalla particolarità dell'accessorio, che ben difficilmente verrà lasciato collegato in permanenza.

Contrariamente all'interfaccia parlante della Cheetah, anch'essa per lo Spectrum, che abbiamo provato sul numero 29 di MC, il prodotto della Currah non dispone di un altoparlante interno, ma manda il segnale sonoro al televisore, miscelandolo al segnale video ad alta frequenza.

In questo modo è possibile ottenere un volume elevato senza dover necessariamente ricorrere ad un amplificatore entrocontenuto, a tutto vantaggio dei costi di produzione. Questa soluzione rende tuttavia inutilizzabile il Microspeech da parte di chi usa come video un monitor al posto del televisore.

La miscelazione avviene all'interno del sintetizzatore, il segnale RF, disponibile sull'uscita TV dello Spectrum, va prelevato con uno dei due cavetti che escono dal contenitore del Microspeech ed il collegamento verso il televisore va effettuato utilizzando l'apposita presa, che si trova in posizione opposta a quella del connettore a 56 poli.

Il secondo cavo, che va inserito nella presa MIC, permette poi di mandare al TV anche i segnali diretti all'altoparlantino interno e, quindi, di ascoltare in maniera finalmente decente gli scarsi effetti sonori ottenibili dallo Spectrum.

Usando il Microspeech ci si accorge subito che l'idea di mandare il segnale audio al televisore, anche se consente una maggiore versatilità, genera qualche problema. La qualità del segnale video infatti peggiora e l'immagine sullo schermo appare afflitta da qualche leggera interferenza.

La prima volta che si installa l'interfaccia può essere necessario procedere ad una breve taratura, volta ad ottenere un suono il più nitido possibile dal TV. Si tratta di un' operazione semplicissima e per nulla critica, che va condotta girando con un cacciavite la vitina di regolazione visibile sul contenitore, fino a trovare il punto in cui l'ascolto è migliore.

Assieme alla confezione vengono forniti due manuali di istruzioni (quello originale inglese e la relativa traduzione italiana) ed una cassetta dimostrativa.

Il manuale è chiaro e piuttosto esauriente, e copre anche gli aspetti più tecnici dell'uso del Microspeech, come il suo pilotaggio da linguaggio macchina.

La cassetta contiene due programmi: il primo è una sorta di mini-corso di apprendimento dell'uso del sintetizzatore articolato in cinque lezioni, mentre il secondo è un gioco di avventura che, ovviamente, non si limita a stampare i messaggi sullo schermo ma li pronuncia (!) ad alta voce

[illegible] collegamenti [illegible] per l'installazione dell'interfaccia [illegible] due cavi: il primo va alla presa MIC, il secondo all'uscita TV.

Caratteristiche ed utilizzazione

Il Microspeech è un sintetizzatore vocale basato sul principio degli allofoni. Se da un lato questa tecnica offre, come già abbiamo accennato, il vantaggio di un vocabolario estesissimo, dall'altro genera un segnale ben lungi dall'essere perfetto e che dà origine ad una voce piuttosto metallica ed abbastanza disturbata da rumore di fondo, pur rimanendo sufficientemente intelligibile. Prima di poter esaminare le modalità d'uso del Microspeech è necessario spiegare brevemente cosa siano gli allofoni sopra citati, per un discorso più ampio vi rimandiamo comunque all'articolo "Analisi e sintesi del segnale vocale", di Leo Sorge, pubblicato sul numero 31.

Gli allofoni possono essere considerati come l'equivalente, per le parole pronunciate, di quello che le lettere dell'alfabeto sono per le parole scritte. Il suono di ogni parola può infatti essere scomposto in un certo numero di "suoni fondamentali", detti appunto allofoni.

Così come le lettere dell'alfabeto, gli allofoni sono in numero limitato (di solito meno di un centinaio) e non sono necessariamente uguali per tutte le lingue: ad esempio in italiano non esiste il suono del gruppo "th" della lingua inglese, d'altro canto in inglese non esiste il suono del gruppo "gl" con la g dolce.

Combinando opportunamente gli allofoni di una lingua è possibile ricostruire il suono di tutte le parole di quella lingua ed è proprio questo il principio di funzionamento del sintetizzatore Currah.

Il set di allofoni disponibile sul Microspeech è, naturalmente, quello dell'idioma britannico, di conseguenza sarà impossibile riprodurre correttamente il suono della totalità delle parole italiane. Nella maggioranza dei casi i risultati ottenibili sono più che buoni, ogni tanto però ci si deve rassegnare a qualche compromesso.

Una caratteristica abbastanza singolare del Microspeech è la sua capacità di pronunciare i significati dei tasti premuti quando si è in modo edit, cioè quando non si sta girando un programma. La possibilità di "ascoltare" ciò che si digita è indubbiamente di aiuto nell'evitare errori di battitura, ma potrebbe rivelarsi molto utile per permettere anche ai non vedenti di utilizzare un calcolatore.

In condizioni normali, tuttavia, la cosa inizia ben presto a dare fastidio e conviene tornare al meno invadente click dando il comando diretto LET keys=0.

Passiamo ora a vedere come si fa per far

Costruttore:
Currah Computer Components Ltd
Distributore per l'Italia
MIPECO casella postale 3016
00121 Roma (Ostia)
Tel. 06/5611251
Prezzo
99.000 lire (comprese IVA e spese postali)

Nella confezione si trovano un [illegible] manuale di istruzioni ed una cassetta contenente un demo ed un programma di avventura.

pronunciare al sintetizzatore delle parole qualsiasi. Il comando da usare, che funziona sia da programma che in modo diretto, è questo:

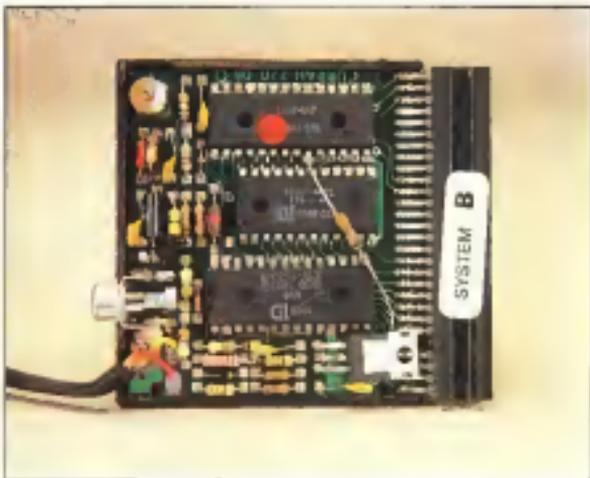
LET s$ = "<sequenza di fonemi>"

Ogni volta che viene assegnata la variabile s$ il programma di gestione del Microspeech ne esamina il contenuto, che deve essere costituito da una sequenza di simboli corrispondenti ad altrettanti allofoni e, se non ci sono errori, inserisce i codici dei relativi allofoni in un buffer situato nella parte alta della Ram. Da qui questi codici vengono prelevati per essere trasformati in suoni e mandati all'audio del TV.

L'uso di un buffer è estremamente importante: grazie ad esso il calcolatore non rimane bloccato quando il Microspeech "parla" e può invece continuare la normale elaborazione. Normalmente la lunghezza del buffer è di 256 byte, e può essere aumentata con facilità abbassando la Ramtop con un CLEAR.

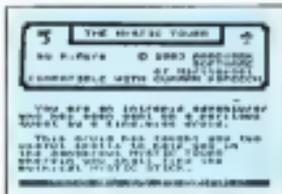
I simboli che rappresentano gli allofoni sono stati scelti in modo che la loro successione sia il più possibile simile alla parola da pronunciare.

Per far dire al Microspeech la parola "hello", ad esempio basta fare

LET s$ = "he(ll)(oo)"

L'hardware comprende tre [illegible] circuiti integrati, un regolatore di tensione, un transistor ed una manciata di componenti passivi.

Nelle occasioni più fortunate, la serie di simboli è esattamente uguale alla parola scritta, un esempio, questa volta in italiano, è il vocabolo "salve"

Se qualche allofono non viene scritto correttamente il sintetizzatore rimane muto, per conoscere dove è stato commesso l'errore si può dare il comando PRINT s$, che stampa la stringa dei simboli segnalando il punto sbagliato con un "?"

Un'apposita tabella del manuale riporta l'elenco di tutti i simboli utilizzabili, affiancati da esempi che mostrano qual è il suono corrispondente a quell'allofono. All'inizio la ricerca degli allofoni che costituiscono una data parola è un'operazione abbastanza lunga, dopo aver fatto pratica con almeno una cinquantina di parole si riesce però

La cassetta dimostrativa contiene anche una simpatica avventura parlante: la Torre mistica

ad andare abbastanza spediti, senza nemmeno la necessità di consultare spesso la tabella degli allofoni

Una possibilità di rilievo, che contribuisce sostanzialmente alla comprensione del parlato e quella dell'intonazione, ogni allofono può infatti essere accentato (l'effetto che si ottiene, in realtà, non è proprio quello di un accento, anche se gli assomiglia molto) scrivendolo in lettere maiuscole

Conclusioni

Il Microspeech Currah è senza dubbio un prodotto molto curato, che si distingue per numerosi particolari che ne rendono l'uso piacevole ed immediato

Ci riferiamo soprattutto alla rappresentazione utilizzata per gli allofoni, all'uso di un buffer che lascia libero il computer durante l'emissione del segnale e all'output attraverso l'audio del televisore

I suoi difetti consistono principalmente nella bassa qualità del parlato riprodotto e nel leggero degrado dell'immagine video, provocato dalla miscelazione tra radio e audio frequenza

Per quello che riguarda il primo bisogna però non perdere di vista la fascia di prezzo di questo accessorio: a meno di 100 000 lire non è veramente possibile fare di meglio

In conclusione si tratta di un add-on simpatico e riuscito, adattissimo per fare qualche esperimento nell'affascinante campo della riproduzione della voce umana, ma anche soltanto per divertircisi un po'

il Personal Computer IBM?

La Libreria Programmi Personal Computer IBM, per esempio.

Il tuo Concessionario IBM Personal Computer ha una novità per te: la Libreria Programmi Personal Computer IBM.
Sulla Libreria sono disposti i principali programmi progettati per il tuo Personal Computer, programmi studiati proprio per rispondere a tutte le tue esigenze di lavoro nel modo migliore.
La Libreria Programmi Personal Computer IBM è uno strumento utilissimo, che ti permette di trovare nel modo più immediato i programmi firmati da IBM. Puoi guardarli e confrontarli per vedere subito, fra quelli di uno stesso settore applicativo, quale sia il programma che risponde meglio alle tue esigenze.
Ma la Libreria Programmi Personal Computer IBM ti dà anche un aiuto in più: con ogni programma troverai la brochure che lo riguarda. Ce ne sono anche molte altre, che ti illustreranno tutto quello che il tuo Personal Computer IBM può fare e le sue utilizzazioni particolari, come il Videotel. Puoi prenderle e portarle a casa, per leggerle in tutta tranquillità e pensare bene alla scelta che devi fare.
E non dimenticare che il Concessionario IBM (gli indirizzi sono sulle Pagine Gialle) è un vero esperto, che conosce perfettamente il Personal Computer IBM ed i problemi della tua attività. Potrà consigliarti nel modo migliore nella scelta dei programmi e ti illustrerà tutte le possibilità che il Personal Computer IBM ti offre, oltre alla grande versatilità delle sue prestazioni.
Per acquisto, consulenza e servizi, puoi anche rivolgerti al Negozio IBM Centromilano.
Allora, non ti pare che il Personal Computer IBM abbia veramente qualcosa in più?

COMPRA
UN PERSONAL COMPUTER IBM
E puoi vincere un fantastico viaggio sulla rotta di Colombo.
I Concessionari IBM e il Negozio IBM Centromilano ti aspettano dal 15/3 al 15/4 1985.

IBM

Aut. Min. Conc.

Paintstar
per Apple II

di Valter Di Dio

Usare le pagine grafiche dell'Apple per disegnare è sempre stato il desiderio di tutti, ma d'altra parte un buon programma grafico costa diversi soldini e spesso necessita di speciali accessori anche questi dal costo non indifferente.

In genere poi le software house tendono a vendere oltre al software, facile preda dei pirati, anche un minimo di hardware che renda inutile la copia del programma.

La Horizontal Line Software invece fornisce solo il programma Paintstar su disco. Il joystick necessario al funzionamento del programma può essere uno qualsiasi già in possesso dei pittore.

La confezione del Paintstar contiene il dischetto del programma e il manuale d'uso, che come riportato dal timbro sulla confezione è in italiano. Il manuale, molto probabilmente realizzato con un Macintosh, è un fascicoletto con riportate tutte le funzioni presenti sul menu, compreso il disegnino associato e una breve descrizione di quello che si può fare con detta opzione.

All'inizio del manuale, dopo la garanzia che promette la sostituzione del disco nel caso si logorasse per l'uso, viene specificata la configurazione minima del sistema. Questo deve essere un Apple IIe o IIc con un Drive e un joystick, e se si desidera una stampante grafica. Per i possessori di un vecchio Apple II europlus è invece necessario aggiungere l'espansione di memoria da 16K o la Language Card, infatti il Paintstar risiede proprio negli ultimi 16K di memoria e utilizza tutto il resto della memoria per contenere il disegno e i dati necessari alle varie opzioni.

Se si dispone della stampante il Paintstar può anche effettuare una copia su carta del disegno corrente. Un secondo Drive, opzionale, si rivela però particolarmente comodo in quanto le immagini non possono essere salvate sullo stesso disco del Paintstar, e del resto, il disco originale deve essere presente nel drive per alcune opzioni: Hard copy, scelta del set di testo e operazioni su file.

Come si usa

Una volta inserito il joystick nella porta del computer e il dischetto Paintstar nel drive basta accendere la macchina per trovarsi davanti il menu principale.

Il menu è particolarmente facile da usare per le piccole immagini che danno subito un'idea del funzionamento dell'opzione selezionata. Sulla destra dello schermo compare il set di 'pennelli', quello selezionato appare in reverse, il pennello di default, il più piccolo, è grande un solo punto, seguono due dimensioni di poco più grandi e quindi due barre, una verticale e una orizzontale molto comode per ampie zone da 'verniciare'. Gli ultimi tre tipi sono già delle piccole figure, possono essere usate sia come pennello, e in tal caso lasciano una scia piuttosto grande, oppure si possono sparare come singole immagini per fare ad esempio le stelline di un albero di natale.

Appena alla sinistra dei pennelli si trova la tavolozza dei colori. È questa un'opzione particolarmente utile, infatti l'Apple II ha di serie solo quattro colori (oltre al bianco e al nero ovviamente) e per fare della 'pittura' sono un po' pochini. Il Paintstar permette di avere 12 (dodici) colori divisi in due gruppi, questa divisione deriva da una caratteristica hardware dell'Apple che non può avere colori appartenenti a gruppi diversi nello stesso byte. Questo spiega la presenza di due Neri e due Bianchi, per evitare problemi conviene infatti usare il bianco o il nero della stessa colon-

Il menu del Paintstar presenta delle piccole immagini dette icone [illegible] dell'opzione. Per selezionarla basta portare [illegible] e premere il tasto sul joystick. Nella foto di destra l'insieme dei colori disponibili con il Paintstar.

na del colore che si stava usando Delle due caselle U1 ed U2 parleremo dopo in quanto fanno riferimento ad una particolare opzione del Paintstar

Le opzioni del menu

La prima opzione da selezionare è la KILL PICTURE (con il disegnino di un teschio) questa cancella il disegno con il colore in uso, oltre che per cancellare si usa quindi anche per definire un fondale su cui lavorare La pagina grafica al lancio del programma è ancora sporca e occorre pulirla prima di iniziare il lavoro Per disegnare si parte di solito con l'opzione Pencil che trasforma il joystick in una matita premendo il tasto si abbassa la matita sul foglio e si possono tracciare delle linee a 'mano libera'

Molto comodo è il fatto che oltre che dal joystick il cursore può essere mosso anche dai tasti I,J,K,M (maiuscole) in modo da poterlo posizionare con precisione in qualsiasi punto La velocità di movimento col joystick può essere variata premendo un numero da zero a nove (0 = veloce) in modo da adattarsi al tipo di disegno che si sta realizzando

Oltre che come una matita il cursore si può usare per tracciare linee rette, in tal caso l'opzione DRAW LINE trasforma la matita in una specie di righello Particolarmente comodo il fatto che, una volta fissato il primo punto con la pressione del tasto una sottile riga mantiene il collegamento con il cursore indicando come sarà la riga finale

Più o meno simile il tracciamento di riquadri (DRAW FRAME) e scatole (BOX) in questo caso tra il primo punto fissato e il cursore compare un sottile riquadro, alla

Produttore:
Horizontal Line Software
Via Dalmazia 2 - 27029 Vigevano (PV)

Distributore:
E M S
European Society of Medical Software
Via Dalmazia 2 - 27029 Vigevano (PV)

Prezzo:
L. 69 900 più IVA

pressione del pulsante di conferma il riquadro si fissa e, se si tratta di una scatola, si riempie in un attimo del colore selezionato

Una cosa strana succede invece se desideriamo tracciare circonferenze Infatti ci aspetteremmo di fissare il centro e vedere il raggio che segue il cursore, invece il comportamento è identico al caso dei riquadri, infatti, per ragioni di velocità il Paintstar traccia la circonferenza o l'ovale inscritto nel riquadro visualizzato Se è così possibile tracciare con un'unica opzione cerchi ed ellissi, è in verità piuttosto difficile tracciare un cerchio perfetto e anche il tracciamento risente un pochino di questa scelta trasformando i cerchi molto piccoli in ottagoni Forse sarebbe stata comoda un'altra opzione per tracciare cerchi perfetti magari a velocità ridotta

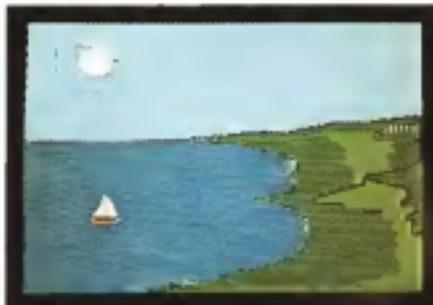
Ultima opzione tra quelle classiche è la PAINT che permette di riempire un'area circoscritta di un colore qualsiasi. Anche questa opzione è stata implementata con una routine particolarmente veloce, tanto che forme strane che occupano quasi tutto lo schermo vengono campite in pochi attimi Un po' di attenzione va prestata però nel suo uso Infatti dovete vedere la funzione Paint come della vernice che viene versata nel disegno se ci sono dei buchi la vernice esce e sporca tutto In pratica, per motivi interni dell'Apple, le uniche linee continue sono quelle bianche (sempre) e quelle dei quattro colori originali solo in verticale Capitano quindi molto spesso fughe di vernice e a questo punto risulta utilissima la prossima opzione

La OOPS permette di annullare l'effetto sul disegno di tutto quello che si è fatto con l'ultima opzione, in pratica si torna col disegno nelle condizioni precedenti all'ultima scelta del menu Questo, eliminando la paura di sbagliare, come al solito fa commettere meno errori e rende più veloce il lavoro

Ultima opzione di disegno vero e proprio e la SPRAY, alla vista di questa opzione mi ero illuso che fosse come la Spray del Macpaint che permette di ombreggiare le figure per ottenere un effetto tridimensionale, purtroppo invece la Spray non fa altro che rendere intermittente la traccia lasciata dal joystick e quindi tanto più rapido è lo spostamento tanto maggiore sarà la distanza tra i punti La Spray si può usare con un po' di esperienza per ombreggiare, ma l'effetto è di solito più disastroso che utile

Per effettuare dei lavori di precisione, ad esempio un'ombreggiatura, è molto meglio usare il Magic Zoom che ingrandisce di sette volte l'immagine (naturalmente una parte di questa) Con lo Zoom è possibile accendere e spegnere singoli punti dello schermo e questo si rivela molto utile in fase di rifinitura del disegno

Le utility

Sempre senza lasciare il menu principale è possibile accedere ad alcune utili opzioni che pur riguardando il disegno, non sono proprio delle opzioni grafiche

[illegible] riempire molto velocemente aree chiuse. La presenza di fori come nel caso della foto di sinistra tra il cielo e la terra lascia uscire il colore che campisce anche la terra (foto centrale). Nella foto a destra il particolare della scritta ottenuto con la funzione Zoom

La prima è la TYPE che permette di scrivere sul disegno usando la tastiera e un set di otto stili che si trovano sul disco. Un grosso limite di questa opzione è il fatto che si può solo scrivere in orizzontale e a grandezza fissa. Se servono scritte particolari o si disegnano punto per punto con il Magic Zoom oppure si salva il disegno senza scritte e si aggiungono poi con un altro programma (ad esempio quello della tavoletta grafica di MC).

La seconda utility è la TAKE OBJ. che permette di copiare parti del disegno in un'altra zona, anche parzialmente sovrapposte. Con la Take obj si possono anche togliere completamente parti del disegno da un posto per portarle in un altro.

La terza utility di disegno è senza dubbio la più originale, permette infatti di definire un 'pattern' e di utilizzarlo come tredicesimo colore, appunto U1 ed U2.

Un pattern è un insieme di punti accesi e spenti che compongono una matrice di 8 punti per otto righe. È proprio con pattern particolari che si ottengono a partire dai sei colori fondamentali tutte le tinte utilizzate nei disegni, questo fa sì che in pratica il numero di colori utilizzabili con il Paintstar sia ben superiore ai dodici visualizzati nel menu. Oltre alle tinte con la User Define Pattern è possibile anche creare quelli che i disegnatori chiamano retini, ossia una serie di righe o quadretti o (volendo) fiorellini con cui riempire delle aree. Proprio con la Def U Pat è stato realizzato il fondale del disegno della tavolozza.

La stampante

Il Paintstar è predisposto per effettuare automaticamente l'Hard copy grafica su tre tipi di stampante: Apple Image Writer, Apple Dot Matrix e Epson FX/RX 80/100 con APL.

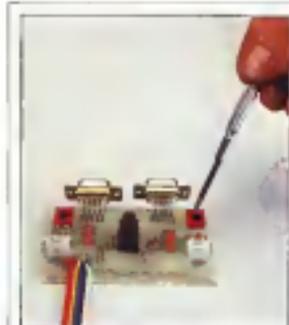
Quando dal menu si seleziona la figuretta dell'Hard copy il disco si mette in movimento e compare il menu di scelta del tipo di stampante, della interfaccia usata e del tipo di copia che si desidera. La copia può essere normale o in inverso e normale o ingrandita due volte.

Purtroppo il programma non memorizza la configurazione sul disco per cui questa tiritera è necessaria ogni volta che si desidera una copia.

Conviene di solito salvare il disegno su un disco e, usciti dal Paintstar, ricaricare i disegni e usare i normali comandi di Hard copy.

Per quanto riguarda il salvataggio su disco del disegno questo avviene come un normale DUMP della pagina grafica (34 settori) con il nome che preferite, si possono quindi preparare immagini sia per programmi che vogliono nomi tipo PIC NOME sia NOME PIC o nomi senza prefissi.

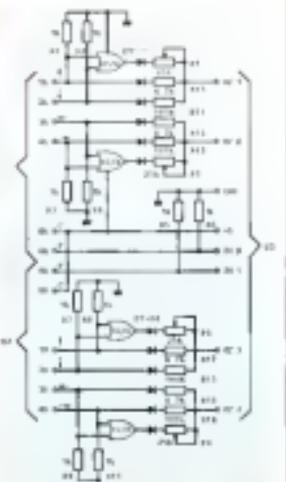
Conclusioni

La principale caratteristica di questo programma è la incredibile velocità di esecuzione, soprattutto delle routine di riempimento di aree, questo permette di preparare un disegno in tempi molto brevi e di poter anche effettuare prove e riprove senza dover attendere svariati minuti ogni volta che si colora una figura. L'uso del joystick come elemento di lavoro si rivela poi oltremodo comodo ed economico, non è neanche necessario usare un joystick originale in quanto il Paintstar controlla solo se la leva si trova al centro o no. Il menu con le icone aiuta enormemente in quanto non occorre decifrare il significato delle abbreviazioni, ma basta cercare la figurina giusta, posizionarcisi sopra e premere il pulsante, insomma la filosofia del Mac sull'Apple II. MC

Joystick economici per Apple II

Dal momento che molti programmi commerciali non sfruttano realmente le possibilità del joystick originale, alquanto costoso, ma considerano solo le otto direzioni della leva senza badare ai valori intermedi, è possibile realizzare, ad un costo modesto, una interfaccia che consente di attaccare alla porta giochi di un Apple due joystick economici sul tipo di quelli che la Spectravideo ha realizzato per il Commodore 64.

Per realizzare l'interfaccia (presentata sul numero 29) bastano pochi componenti e un minimo di pratica con il saldatore, per semplificare il lavoro e ridurre la possibilità di errori abbiamo preparato un circuito stampato con la serigrafia dei componenti, ordinabile presso la redazione al prezzo di 9500 lire IVA compresa.

Elenco componenti

da R1 a R10	=	1 kohm ¼ watt
da R11 a R14	=	100 kohm ¼ watt
da P1 a P4	=	25 kohm lineare
da D1 a D12	=	1N4148
IC1	=	SN 74 LS 02
U1 e U2	=	Cannon maschi a 8 poli
U3	=	zoccolo per I C a 16 pin

Schema elettrico dell'interfaccia Joystick

PER IL TUO COMPUTER
GIOCHI E UTILITY SU CASSETTA!
Se hai lo
spectrum
in edicola
Elettronica 2000
l'hardware
Raccolta Speciale
NUOVA!
commodore 64
UNA FANTASTICA COMPILATION

DELTA: IL DATABASE DI FIDUCIA!

Volete gestire le informazioni in modo facile, veloce ed accurato? Il database DELTA è la sola risposta a tutti questi attributi. DELTA vi permette di creare qualsiasi tipo di record e soddisfare le vostre esigenze!

DELTA è un generatore di applicazioni ed è ideale per archivi del personale, ospedalieri, agenzie viaggio, libreria, magazzino, fatturazione, laboratorio, studio avvocati e mille altri!

DELTA è completamente in ITALIANO (manuale e dischetti). E'facilissimo da usare! Non richiede la conoscenza di linguaggi di programmazione. Il menu principale guiderà l'utente passo dopo passo. Ideale per l'uomo d'affari, DELTA è usato ogni giorno anche da esperti programmatori in piccole o grandi aziende in tutta Europa!

Con DELTA è possibile selezionare, classificare, elaborare, produrre prospetti, etichette ecc. e persino inviare lettere personalizzate! DELTA è capace di trasferire dati su/da altri programmi quali Multiplan, Lotus 1-2-3, Wordstar ecc.

DELTA è in grado di ricordare una serie di diverse operazioni e ripeterle poi quante volte lo si desideri. L'utente potrà così creare il proprio menu di opzioni automatiche!

di Francesco Petroni

Ricominciamo da capo Elementi fondamentali

Continuiamo la serie di articoli intitolati "ricominciamo da capo" trattando altri problemi fondamentali in Computer Grafica destinati principalmente ai principianti

Un primo gruppo di problemi, oltre ad essere propedeutico a quasi tutti i temi di Computer Grafica, e anche elementare al punto che gli esempi che pubblichiamo a corredo dell'articolo, sono dei programmi lunghi una sola riga. Questo primo gruppo e costituito da

— programmi che si basano su uno/due loop, oppure non si basano su loop,

— programmi in cui occorre stabilire un punto iniziale, in pratica il punto di partenza del disegno che costituisce l'output, oppure pragrammi in cui il punto di partenza non è identificabile,

— programmi grafici che contengono condizioni logiche (salti condizionati o espressioni logico-matematiche)

Vedremo come risolvere, se è possibile farlo, il problema della eliminazione dei salti.

Analizzeremo inoltre, nel corso dell'articolo, un'altra problematica classica della tecnica grafica e di conseguenza della Computer Grafica, la scelta della Scala di Rappresentazione di un output.

La possibilità, offerta dal Basic interprete di quasi tutti i microcomputer, di poter scrivere più istruzioni sulla stessa riga trova in genere due limiti: il limite fisico della lunghezza massima della riga (al massimo 254 caratteri) e un limite logico dovuto al fatto che non è possibile utilizzare istruzioni di salto dirette all'interno di una riga.

Ciascun programmatore ha un suo stile di lavoro. C'è chi cerca di compattare il programma al massimo, evitando REM, utilizzando tabelle, inzeppando le righe di istruzioni con l'obiettivo di risparmiare memoria e quindi di velocizzare l'esecuzione, ma raggiungendo anche lo scopo di rendere il listato poco "leggibile" agli altri e a se stesso.

C'è, dal lato opposto, chi diluisce il listato al massimo, inserendo REM che commentano non solo ogni subroutine, ma ogni singola istruzione, o che addirittura hanno scopo decorativo. Il risultato è quello di avere listati chilometrici dove però le righe interessanti sono pochissime.

L'atteggiamento intermedio è, come al solito, il più logico e consiste nel cercare il giusto equilibrio fra l'economia di memoria, che è una delle poche regole fondamentali del programmatore, e la leggibilità dei listati. E leggibilità significa facile individuazione delle varie routine e comprensione del loro svolgimento.

L'idea di fare programmi lunghi una sola riga non aggiunge nulla a quanto detto, ma dati gli illustri precedenti (vedi articolo di B.A. sul numero 3 di MCmicrocomputer) mi sento stimolato a provarci finalizzando questi tentativi a programmi di argomento grafico.

Vediamo quindi di elencare e descrivere le varie fasi di un programma generico (grafico o no) e di specificare quali siano le problematiche connesse alle varie fasi, in funzione del voler realizzare un programma di una sola riga e del fatto che tale programma deve essere grafico.

Un qualsiasi programma ha una fase iniziale, nella quale vanno specificate, una volta per tutte, le costanti e definiti i valori delle variabili. C'è poi la fase centrale (che definiremo MAIN Program) in cui vengono svolte le elaborazioni principali. C'è poi la fase finale, che può comprendere l'esposizione del risultato, oppure semplicemente la fine dell'elaborazione, che avviene a causa del verificarsi di una data condizione. Sono considerazioni molto ovvie, ma che è bene aver chiare quando si programma.

Anche un programma di disegno, per quanto semplice, ha questa struttura. Se si disegna una circonferenza, ad esempio, occorre predefinire centro e raggio, ma poi occorre stabilire il punto dal quale partire, il "senso di marcia" e definire la condizione di fine, corrispondente al tracciamento di un intero angolo giro.

Se il Basic possiede l'istruzione CIRCLE il problema per il programmatore non esiste, in quanto la soluzione è insita nell'istruzione CIRCLE. Ma in generale occorre stabilire le varie fasi del disegno (inizio, svolgimento, fine) così come quando usiamo la matita cominciamo da un punto e finiamo con un altro punto.

Anche le istruzioni tipiche del Basic grafico seguono questa logica.

Vi sono istruzioni di inizio linea:

```
PSET (X,Y) per il Basic IBM/Olivetti
HPLOT (X,Y) per l'Applesoft
```

Istruzioni di continuazione o di fine linea:

```
LINE -(X,Y) e HPLOT -(X,Y)
```

Istruzioni che iniziano e finiscono una linea:

```
LINE (X,Y)-(X1,Y1) e HPLOT (X,Y)-(X1,Y1)
```

Nel programma occorrerà individuare una o più condizioni di inizio e una o più condizioni di fine, e conseguentemente occorrerà utilizzare questa o quella istruzione.

In pratica però la scelta di una soluzione rispetto alle altre rimane compito di chi fa il programma.

Per fare un esempio supponiamo di dover tracciare la traiettoria di un oggetto che si muova con una certa legge matematica. Abbiamo almeno tre soluzioni.

Innanzitutto dobbiamo calcolare varie posizioni assunte dall'oggetto e cioè quella iniziale (corrispondente al momento del lancio), quella finale (corrispondente al raggiungimento del bersaglio o all'uscita dal formato video) e un certo numero di posizioni intermedie. Il numero di calcoli intermedi dipende dalla precisione, che vogliamo nel tracciato con l'ovvia considerazione che una maggior precisione costa maggiori tempi di elaborazione.

Le tre soluzioni sono:

— precalcolo delle posizioni (in formato video), loro memorizzazione in una tabella e loro successiva visualizzazione. Il risultato consiste nel disegnare una serie di segmentini, ciascuno dei quali ha in comune con il precedente e il successivo il punto iniziale e il punto finale. La condizione di inizio è individuata dal primo elemento della tabella e quella di fine dall'ultimo.

— Calcolo e visualizzazione immediata di ciascuna posizione. In tal caso occorre individuare il punto iniziale (ad es. se la funzione è legata al tempo occorre individuare la posizione al punto zero) e il punto finale (ad es. quando le coordinate correnti corrispondono a quelle del bersaglio). L'inconveniente è che se le due precedenti condizioni sono individuate a mezzo istruzione IF, occorre fare i due test anche per tutte le posizioni intermedie, con grave rallentamento dell'esecuzione. Il test per il punto iniziale si può evitare facilmente isolandone il calcolo e la visualizzazione.

— Calcolo di due posizioni successive e visualizzazione del segmentino che le unisce. Questo metodo ha il vantaggio di utilizzare una sola istruzione grafica e cioè DA   A. Lo svantaggio consiste nel fatto che ciascun punto deve essere calcolato due volte oppure i valori calcolati dovendo

essere usati due volte debbono essere passati da una variabile (quella di fine segmento) a un'altra (quella di inizio segmento)

Va poi considerata la differente logica delle due condizioni iniziale e finale. Mentre la prima è sicuramente identificabile e quindi isolabile, la seconda si può presentare in varie forme, ad es

— Programma che non finisce mai. Quando, volutamente o meno, si creano delle iterazioni interrompibili solo con un Break.

— Programma che finisce in seguito al verificarsi di una condizione. Ovvero all'interno della elaborazione principale esistono uno o più test logici per verificare il raggiungimento di una condizione finale (es. il raggiungimento del bersaglio).

— Programma che finisce con la fine dell'esecuzione del Main (es fine di un loop)

Ciascuna di queste forme ha un suo campo di applicabilità

In definitiva si può dire che non esiste una regola di comportamento migliore delle altre e quindi la scelta della soluzione da adottare tra le varie possibili è affidata alla sensibilità del programmatore

Istruzioni di salto e di salto condizionato

Le istruzioni di confronto e quelle analoghe di salto condizionato sono di importanza fondamentale nella programmazione in qualsiasi linguaggio e su qualsiasi argomento, a tal punto che non esisterebbero né programmi né probabilmente computer se non si potesse eseguire queste funzioni

E come tutte le istruzioni, permettono sia un uso standard, quello citato nei manuali e che deve essere ben noto a chi programma, sia un uso "avanzato" che sfrutta al massimo anche le possibilità "nascoste" dell'istruzione, e che quindi consente soluzioni brillanti per problemi complessi. Questo vale anche per i programmi grafici dove, come abbiamo visto, esistono sempre problemi di individuazione e trattamento di condizioni particolari

Le istruzioni sono:

IF >condizione< THEN >istruzioni< A<ELSE>istruzioni B<ON>condiz numerica<GOTO/GOSUB>riga di programma <WHILE>condizione numerica< >istruzioni<WEND ecc

Quando la condizione è riducibile a una semplice espressione logica (es M > 10 oppure A$ = "FINE") si possono usare espressioni matematiche contenenti espressioni logiche

Infatti un confronto del tipo citato produce un risultato numerico (1 se la condizione è vera -1 (oppure 0 per altri computer) se la condizione è falsa) e tale risultato può essere inserito come elemento di una espressione matematica complessa.

In figura 1 vediamo l'esecuzione di una serie di esperimenti su tale problematica. Data la semplicità con la quale si possono eseguire esercizi usando il Basic in modo diretto, è opportuno oltre allo studio dei manuali, da ritenere "obbligatorio", anche molto esercizio, specialmente se questi argomenti che vanno assolutamente padroneggiati da chi programma

Programma dodici settori

La modalità SCREEN 3 disponibile nel GWBASIC dell'Olivetti M24 offre uno schermo di lavoro di ben 256 000 pixel E poiché questa modalità è presente anche sul monitor B&N si può dire che questa,

```
print 3>3,3=3,3<3
 0              1               0
print 3*(3=3),3*(3=3+1)
-3              0
print 3*((3=3)+1),3*(3=(3+1))
 0              0
a=4
b=5
print a=a,a=a=a
-1              0
print a*(a=a),(a a)=a
-1              0
print b*(b>a)+a*(b<a)
-5
```

Figura 1 - Esercizio sugli Operatori Logici. È un hard copy di alcune operazioni eseguite in IBM Basic usato in maniera diretta (cioè senza programma)

```
100 K=K+S:PRINT TAB(K) ▮;IF K 10 THEN S= 1:GOTO 100,ELSE IF K 1 THEN S=1:GOTO 1
00: ELSE GOTO 100
101 REM
199 REM
200 K=K+S:PRINT TAB(K);:IF K 12 THEN S=-1:PRINT " :GOTO 200:ELSE IF K 1.1 THEN
S=1:PRINT  ":GOTO 200: ELSE IF S= 1 THEN PRINT "/":GOTO 200:ELSE PRINT "\ :GOTO
200
 201 REM
299 REM
300  LS:S=1:Y=1:FOR K=1 TO 79:Y Y+S:T 20:Y+1)+1+(Y 1 AND Y 24)- Y=24:S=S T:LOCA
TE Y,X:PRINT "@";:NEXT X
```

Figura 2 - Tre programmi Monoriga Alfanumerici su IBM. La sequenza di caratteri in output viene gestita dai vari statement inseriti in ciascun programma

```
100  PRINT  CHR$ (7)  REM   GRIGLIA
110  HGR2   HCOLOR= 3 P = 9 A = A B = 279 C = B D = 191 S = B / D  FOR
     I = A TO B STEP P  HPLOT I C TO I D  HPLOT A I / S TO B,I / S  NEXT
     I
200  PRINT  CHR$ (7)  REM   RAGGIERA
210  HGR2   HCOLOR= 3 P = 9 A = 0 B = 279 C = 0 D = 191 S = B   D  FOR
     I = A TO B STEP P  HPLOT I C TO B - I D  HPLOT A I / S TO B D -
     I / S  NEXT I
300  PRINT  CHR$ (7)  REM   VOLUTA
310  HGR2   HCOLOR= 3 P = 9 A = A B = 279 C = B D = 191 S = B / D  FOR
     I = A TO B STEP P  HPLOT I I / S TO B - I D  HPLOT B - I I   S TO
     A D - I / S  NEXT I
400  PRINT  CHR$ (7)  REM   SPIRALE
410  HGR2   HCOLOR= 3 P = 3 14159  HPLOT 140 96  FOR A = 0 TO 8 * P STEP
     P   15 P = R + 1 X = 140 5 + P *  COS (A) Y = 96 5 +  7 * P *  SIN
     (A)  HPLOT  TO X,Y  NEXT A
500  PRINT  CHR$ (7)  REM  DIAMANTE
510  HGR2   HCOLOR= 3 P = 3 14159 P = 139  FOR A = 0 TO 2 * P STEP P
        15 X = 140 5 + R *  COS (A) Y = 96 5 +  684 * R *  SIN (A)  HPLOT
     0 0 TO X Y  HPLOT  TO 279 191  HPLOT 279 0 TO X,Y  HPLOT  TO 0,
     191  NEXT A
600  PRINT  CHR$ (7)  REM  TRATTEGGIO
610  HGR2   HCOLOR= 3 P = 9 B = 279 D = 191 S = B / D  HPLOT 0,0 TO
     0 0 TO B,D TO 0,0 TO 0,0  FOR I = B TO 279 L =  INT (I / 31) *
     31  HPLOT L D - I / S  NEXT I
700  PRINT  CHR$ (7)  REM   SCALETTA
710  HGR2   HCOLOR= 3 P = 9 B = 279 D = 191 S = B   D  HPLOT B B TO
     0 0 TO B,D TO 0 D TO 0,0  HPLOT 0 D  FOR I = 0 TO 279 L =  INT
     (I / 31) * 31  HPLOT  TO L D - I / S  NEXT I
800  PRINT  CHR$ (7)  REM  POLIGONI
810  HGR2   HCOLOR= 3 P = 3 1416 S = 2 * P -  001  FOR L = 3 TO 8 R =
     L * 11  HPLOT 140 5 + R,95 5  FOR A = 0 TO 2 * P STEP S / L X =
     140 5 + R *  COS (A) Y = 95 5 + R *  SIN (A)  HPLOT  TO X,Y  NEXT
     A  NEXT L
900  PRINT  CHR$ (7)  REM  CICLOIDE
910  HGR2   HCOLOR= 3 P = 3 1416 S = P / 20 P = 2 111  HPLOT 170 95
      FOR A = 0 TO 10 * P STEP S K = P * A X = 140 5 + 57 *  COS (A)
      - 27 *  COS (K) Y = 95 5 + 57 *  SIN (A) - 27 *  SIN (K)  HPLOT
      TO X,Y  NEXT A
1000  PRINT  CHR$ (7)  REM  MERLETTO
1010  HGR2   HCOLOR= 3  DIM X(50) Y(50) P = 6 283   19  FOR I = 1 TO
     19 A = P * I X(I) = 130 *  COS (A) + 140 Y(I) = 90 *  SIN (A) +
     96  NEXT I  FOR L = 1 TO 19  FOR I = L TO 19  HPLOT X(I) Y(I) TO
     X(L) Y(L)  NEXT I,L
```

Figura 3 - Dieci programmi Monoriga Grafici su Apple. Alcuni programmi sono una versione "spartana" dei vari programmi apparsi più volte in questa rubrica

insieme alla velocità dell'8086, doppia rispetto a quella dell'8088, sono due caratteristiche con le quali si supera un bel po' lo standard IBM

Il programma, listato in figura 7 e il cui output è riportato in figura 8, divide lo schermo in 12 settori in ciascuno dei quali disegna una certa curva bidimensionale, che viene anche descritta con un titolo.

Poiché tutte le curve sono del tipo Y = FUN Z(X) è stato possibile inserire ciascuna curva in una sua riga, nella quale è stata inserita anche la scritta sotto forma di DATA

L'esecuzione vera e propria viene affidata ad un loop sulla X, eseguito dodici volte per ognuna delle quali viene richiamata la routine della relativa funzione

Non è un programma in una sola riga, ma viene utilizzato un metodo drastico di riduzione spazio, mettendo in evidenza (come direbbe un matematico) tutte le fasi comuni ai dodici segni. Inoltre viene ottenuta la leggibilità del listato in quanto, come detto, ciascuna funzione ha una sua riga e la sua descrizione

In realtà la vera caratteristica del disegno è rappresentata dall'uso delle istruzioni GET e PUT, con le quali viene realizzata una "animazione" in fase di costruzione del disegno, animazione ovviamente non visibile in una foto. Infatti ogni singolo settore viene disegnato nel quadrante in alto a sinistra, poi viene immagazzinato in un ARRAY, con la istruzione GET (X,Y) - (X1,Y1),D%, e poi viene spostato nel quadrante di destinazione, tramite la istruzione, complementare alla precedente PUT (X,Y),D%.

Con questa modalità, che tra l'altro ha un effetto "spettacolare" si semplifica il problema del riferimento in quanto ciascun disegno viene realizzato nel quadrante, di 156 per 128 pixel in alto a sinistra, senza dover effettuare costose operazioni di scaling

Un'ultima notazione riguarda l'uso della Funzione VIEW di riga 140. Con questa funzione si "isola" il quadrante e le successive operazioni grafiche riguardano solo l'interno della VIEW. In questo modo viene pulito il quadrante prima di un nuovo disegno. Il successivo VIEW, di riga 220, serve per "resettare" la finestra e per permettere quindi l'animazione

Programmi di una sola riga

I programmi lunghi una sola riga sono 16. I primi tre realizzati in Basic-A IBM con uscita sul video monocromatico, facilmente adattabili a tutte le altre situazioni (listati contemporaneamente in fig 2). Poi ce ne sono 10 in Applesoft, con uscita su pagina grafica HGR2, listati in figura 3. Infine gli ultimi 3 sono realizzati in GWBASIC con Olivetti M24, che permette in modalità SCREEN 3, una definizione di 640 per 400 pixel. I listati sono in figura 4. Di questi ultimi programmi sono riportati due output in figure 5 e 6

```
100 SCREEN 3:CLS:LINE (0,0) (639,399),,B:X 10.Y 10:PSET (X,Y):FOR X=10 TO 630:Y=
Y 310*((X 320)+1):LINE (X,Y):NEXT X
101 REM
199 REM
200 SCREEN 3:CLS:LINE (0,0) (639,399),,B:X=10:Y=10:PSET (X,Y):FOR X=10 TO 630:X1
=INT(X/40):Y=X (X1*40 10)*((X (X1*40))+1):LINE (X,Y):NEXT X
201 REM
299 REM
300 SCREEN 3:CLS:LINE (0,0) (639,399),,B:FOR I 20 TO 320 STEP 50:X=10:Y=10+I:PSE
T (X,Y):FOR X=10 TO 630:X1=INT(X/40):Y=  X +X1*40 10)*((X (X1*40))+1):LINE (X,Y)
:NEXT X:NEXT I
```

Figura 4 - Tre programmi Grafici Monoriga su Olivetti M24. Mischiando i concetti di loop, operatore logico ecc si ottengono risultati anche complessi, pur nel limite monoriga

Figure 5-6 - Output del programma Olivetti n 3. Il "dente di sega" è realizzato inserendo nella formula matematica un fattore logico, che come visto nel testo può assumere solo due valori

I primi tre hanno una uscita alfanumerica. In quello di riga 100 il carattere "■" viene printato a zig zag sul video. L'inversione del senso è realizzata invertendo il segno delle variabile S, che rappresenta l'incremento della posizione orizzontale del cursore. L'inversione è gestita tramite due IF THEN ELSE nidificati

Il programma di riga 200 è una evoluzione del precedente. In pratica viene riconosciuto il senso corrente o il momento della inversione, e di conseguenza viene stampato un differente carattere. In questo caso la nidificazione degli IF/THEN/ELSE è più complessa.

L'ultimo programma della prima serie (riga 300) utilizza invece espressioni matematiche che contengono fattori logici. In pratica viene stampato il carattere "$". C'è un loop sulla X, che rappresenta la posizione orizzontale del cursore. La posizione verticale Y è calcolata con una espressione che comunque la riporta entro il formato video

Il secondo gruppo di programmi, quelli in Applesoft, sono grafici e quindi realizzano ciascuno un disegno descritto nel titolo del programma stesso. Senza commentarli uno per uno diremo che si basano tutti su uno o due loop, con i quali vengono fatti

```
100 REM dodici curve sul piano
110 REM inizializzazione generale
120 P=3.14159:DIM D%(2800):SCREEN 3:CLS
130 FOR V=1 TO 0 STEP -1:FOR O=3 TO 0 STEP -1:J=(V*4+O+1):K=13-J
140 VIEW (0,0)-(156,128):CLS
150 O1=O*160:O2=O1+156:V1=V*132:V2=V1+128:O3=78:V3=64
160 READ K$:LINE (0,0)-(156,128),,B:LINE (1,1)-(155,127),3,,F
170 LINE (0,64)-(156,64):LINE(78,0)-(78,128):LOCATE 8,2:PRINT J;" "K$;
180 FOR X= 75 TO 75 STEP 2
190 ON K GOSUB 300,310,320,330,340,350,360,370,380,390,400,410
200 X1=X+O3:Y1=Y+V3:IF X=-75 THEN GOSUB 250 :ELSE GOSUB 260
210 NEXT X:IF K= 12 THEN PRINT CHR$(7):GOTO 240
220 VIEW (0,0)-(639,399):GET (0,0)-(156,128),D%:PUT (O1,V1),D%
230 PRINT CHR$(7):NEXT O:NEXT V
240 I$=INKEY$:IF I$="" THEN 240 ELSE SCREEN 0:CLS:END
250 PSET (X1,Y1):RETURN
260 LINE -(X1,Y1):RETURN
300 Y=50*SIN(X/10)/(X/10+.0001):RETURN:DATA "Y=SIN(X)/X"
310 Y=50/X:RETURN:DATA                        "IPERBOLE"
320 Y=.2*X^2-X+2-30:RETURN:DATA               "PARABOLA 2"
330 Y=45*SIN(X/30):RETURN:DATA                "SINUSOIDE"
340 Y=.6*X+12:RETURN:DATA                     "RETTA"
350 Y=20*SIN(X/20)-30*COS(X/24):RETURN:DATA   "TRIGON. 2"
360 Y=10*LOG(ABS(X/80)):RETURN:DATA           "LOGARITMO"
370 Y=20*SIN(X/2)-16*COS(X/3):RETURN:DATA     "TRIGON. 1"
380 Y=10*LOG((X+76)/10):RETURN:DATA           "LOGARITMO"
390 Y=20/SIN(X/10):RETURN:DATA                "SIN INVERSA"
400 Y=(ABS(X/8))^1.2:RETURN:DATA              "ESPONENZIALE"
410 Y=.1*(X/10)^3-.1*(X/10)^2+2:RETURN:DATA   "PARABOLA 3"
```

Figura 7 - Programma grafico in 12 settori su Olivetti. La definizione disponibile (in modo SCREEN 3) è di ben 640 per 400 pixel e la velocità è quella dell'8086. E queste sono condizioni che si sentono

```
100 CLS:REM operazioni manuali di scaling
110 REM X0,Y0 vertice in alto a sinistra finestra video
120 REM X1,Y1 vertice in basso a destra  finestra video
130 REM X8,Y8 vertice in alto a sinistra del soggetto
140 REM X9,Y9 vertice in basso a destra  del soggetto
150 INPUT "Immetti x0,y0  ",X0,Y0
160 INPUT "Immetti x1,y1  ",X1,Y1:PRINT
170 INPUT "Immetti x8,y8  ",X8,Y8
180 INPUT "Immetti x9,y9  ",X9,Y9:PRINT
190 VX=X1-X0:VY=Y1-Y0
200 SX=X9-X8:SY=Y9-Y8
210 FX=VX/SX:FY=VY/SY
220 TX=X0-X8*FX:TY=Y0-Y8*FY
230 PRINT "Form. Video      ";VX;" in Orizz. ";VY;" in Vert."
240 PRINT "Form. Soggetto   ";SX;" in Orizz. ";SY;" in Vert."
250 PRINT "Fatt. Scala      ";FX;" in Orizz. ";FY;" in Vert."
260 PRINT "Fatt. Traslaz.   ";TX;" in Orizz. ";TY;" in Vert."
270 DEF FNXX(X)=X*FX+TX:DEF FNYY(Y)=Y*FY+TX
280 PRINT:INPUT "Punti Soggetto   ",X,Y
290 PRINT "Punti Video      ",FNXX(X),FNYY(Y):GOTO 280
```

```
Immetti x0,y0  100,100
Immetti x1,y1  200,200

Immetti x8,y8  -1, 1
Immetti x9,y9  1,1

Form. Video      100  in Orizz.  100  in Vert.
Form. Soggetto   2  in Orizz.  2  in Vert.
Fatt. Scala      50  in Orizz.  50  in Vert.
Fatt. Traslaz.   150  in Orizz.  150  in Vert.

Punti Soggetto   0,0
Punti Video                150            150

Punti Soggetto    1,1
Punti Video                100            200

Punti Soggetto   1, 1
Punti Video                200            100

Punti Soggetto
```

Figura 9 - Esercitazioni di scaling manuale. Listato e output. La conoscenza dei "sistemi di riferimento" e delle funzioni di "scaling" è un prerequisito fondamentale per chi vuole realizzare programmi grafici.

variare o coordinate (orizzontali e/o verticali) oppure angoli.

Nel programma "SCALETTA" l'effetto scala è generato utilizzando la funzione INT, che permette di tenere fisso il valore Y entro un certo intervallo della X.

Il Basic dell'Apple II non contiene l'IF THEN ELSE. Quindi non è possibile in certi casi lavorare su una sola linea, in quanto la situazione ELSE va riportata alla linea successiva.

L'ultimo gruppo di programmi, scritti in GWBASIC, comprende espressioni logico matematiche abbastanza complesse con le quali si possono ottenere svariati effetti "a basso costo". Ne riportiamo due esempi, anche questa volta senza descrivere il programma nel dettaglio.

È evidente che tali programmi hanno solo uno scopo didattico in quanto praticamente non è mai necessario porsi vincoli così drastici in sede di programmazione. E inoltre lo stesso risultato si può raggiungere con routine più facili da realizzare e più leggibili.

Lo scaling

Un altro elemento base in tutti i problemi di computer grafica è lo "scaling", ovvero quel complesso di operazioni logico-matematiche che vanno eseguite sui dati relativi all'oggetto da visualizzare per tradurli in dati compatibili con i formati in uscita.

È una problematica che entra in tutte le applicazioni e che quindi deve essere ben conosciuta da chi vuol fare in proprio computer grafica senza limitarsi ad usare programmi altrui.

Ai neofiti della computer grafica suggeriamo quindi di studiare il problema e di fare su di esso esercizi per arrivare a padroneggiarlo, cosa che gli renderà più semplici i passi successivi.

Ovviamente il problema è stato più volte trattato in questa rubrica (vedi per esempio MCmicrocomputer n. 3 pag. 52).

Lo "scaling" può essere paragonato alla scelta delle apparecchiature e delle condizioni con le quali fare una fotografia di un certo soggetto. La fotografia può andare dalla microfotografia (apparecchio collegato ad un microscopio), macrofotografia (foto di soggetti molto piccoli e utilizzo di obiettivi particolari) e così via fino alla foto fatta applicando un dorso fotografico ad un telescopio.

Ma anche fotografia di qualche cosa di astratto come ad esempio di una superficie geometrica nello spazio derivante da una formula matematica, soggetto che nessuna macchina fotografica può riprendere.

I passi logici che bisogna percorrere per riprodurre un soggetto con un computer sono:

— esprimere il soggetto in termini di dati matematici; (scelta e preparazione del soggetto),

— sottoporre i dati matematici all'operazione di scaling, (scelta della lunghezza focale dell'obiettivo, per mettere in primo piano il soggetto e puntamento verso il soggetto, che non è assolutamente detto che stia davanti all'obiettivo),

— visualizzare i dati così trattati, (equivalente allo scatto).

Detto in altri termini ciascun dato relativo al soggetto del disegno, può essere traslato, ruotato, ingrandito, rimpicciolito, prima di poter essere visualizzato. E queste operazioni è bene farle subito prima della visualizzazione in quanto riguardano questa fase della procedura (strettamente dipendente dai formati in uscita) e non ad esempio la definizione dei dati del soggetto.

La suddivisione in tre fasi riguarda i tipici programmi di computer grafica. Esiste però la possibilità di evitare una delle fasi. I programmi di una sola riga pubblicati nella prima parte dell'articolo, ad esempio, riguardano dati già inizialmente compatibili con il formato in uscita in modo da rendere inutile lo scaling.

È quindi evidente che tali programmi non "girano" in computer che hanno diffe-

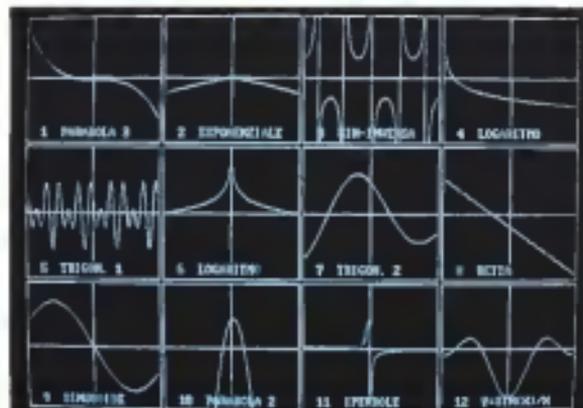



Figura 8 - Output del programma Olivetti. La foto fa perdere l'effetto "animazione". Ogni settore ha una definizione 213 per 133 pixel, più che sufficienti per disegni bidimensionali di una certa complessità.

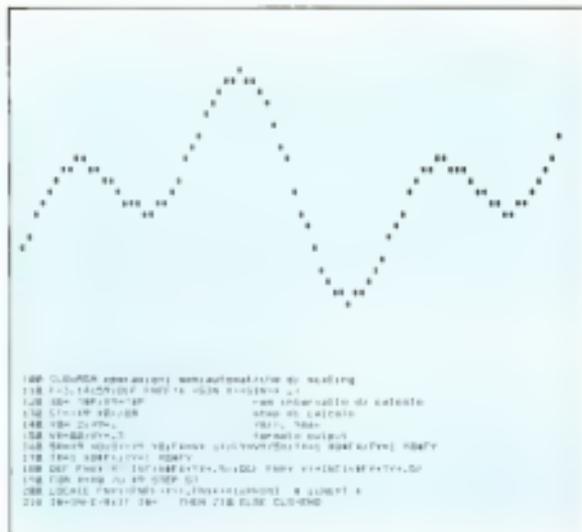



Figura 10 - Scaling semiautomatico. Il programma utilizza come output un video alfanumerico standard. Il formato in uscita è quindi di 80 per 24 punti

renti formati in uscita, in tal caso si fa lo scaling oppure si cambiano i dati iniziali.

Come si fa lo scaling?

Bisogna considerare tre elementi:

1 - si vuole/non si vuole fare lo scaling automatico.

Cioè vogliamo imporre dei margini "esterni" rispetto al soggetto oppure deve essere il programma che calcola i margini del disegno automaticamente in modo da riempire il più possibile il formato di uscita.

Ad esempio se di alcuni valori statistici vogliamo costruire un istogramma, uno scaling automatico porrà il massimo della scala di riferimento in funzione del massimo valore da visualizzare. Uno scaling manuale, viceversa, impone una scala di riferimento indipendentemente dai valori delle colonnine.

2 - Esistono/non esistono funzioni di scaling nel linguaggio.

Alcuni Basic più avanzati comprendono istruzioni che eseguono uno scaling automatico consistente nel poter imporre un certo formato di uscita dipendente dal soggetto. Ad esempio se vogliamo visualizzare una funzione in un intervallo (—1, +1) della X, entro il quale i valori Y oscillano tra -2 e 2, basta usare l'istruzione WINDOW (-1, -2) - (1, 2) e automaticamente viene eseguito lo scaling che traduce i valori interni a tali intervalli nei valori del formato massimo della macchina.

3 - Si conoscono/non si conoscono subito tutti i dati da visualizzare.

Se il soggetto è il risultato di un calcolo complesso e probabile che i dati da visualizzare non si conoscano se non dopo la elaborazione principale e poiché lo scaling si fa, come vedremo, considerando i valori massimi e minimi di tali dati occorreranno tre successive elaborazioni, la prima di predisposizione dei dati e di ricerca dei valori max, min, la seconda di definizione delle funzioni di scaling, la terza di scaling di tutti i dati calcolati precedentemente.

Le funzioni classiche di scaling sono, nel caso non vi siano rotazioni:

X2 = X1 × S + TX	X2, Y2	dati in formato out
Y2 = Y1 × S + TY	X1, Y1	dati in formato iniziale
	S	fattore di scala
	TX, TY	valori di traslazione

nel caso invece vi sia una rotazione (angolo tra i due riferimenti A),

X2 = (X1 × COS(A) + Y1 × SIN(A)) × S + TX
Y2 = (-X1 × SIN(A) + Y1 × COS(A)) × S + TY

infatti per A=0 si hanno le funzioni di prima.

Gli esercizi

Il primo programma di esercitazione non produce output grafico, ma un output alfanumerico dei risultati di una operazione di scaling. Vengono chiesti dapprima il formato output (attraverso i vertici della diagonale) e poi il formato massimo del soggetto (sempre attraverso la diagonale del rettangolo). Con questi valori si calcolano i range delle X e Y rispettivamente del formato dati e del formato out (righe 190 - 200) e i valori degli spostamenti TX, TY necessari per "centrare" il disegno (210). Con tutti questi valori vengono costruite due funzioni FNXX(X), FNYY(Y) da utilizzare nella routine di traduzione (riga 270), che richiede in INPUT coppie di punti in formato dati che vengono restituiti in formato output (righe 280-290).

Il formato di uscita dipende dalla definizione, in pixel, dell'uscita grafica del computer, ma di questa uscita se ne può utilizzare anche una porzione, ad esempio se si vuole fare un disegno all'interno di una finestra. Il formato massimo dei dati è equivalente al rettangolo che ha come vertici della diagonale i punti ideali (Xmin, Ymin) e (Xmax, Ymax), ideali perché non è detto che esista un punto Xmin, Ymin da visualizzare.

L'esempio pubblicato assieme al listato (in fig. 9) chiarisce un po' di più il funzionamento dell'operazione "scaling".

Il secondo programma viene definito "scaling semiautomatico" in quanto prevede un'uscita su video alfanumerico standard (di 80 per 24 caratteri), prevede la definizione a programma del range di calcolo della funzione (da -3 pigreco a +3 pigreco) e del range della Y in output (da -2 a +2).

In pratica le funzioni di scaling, definite in riga 180, traducono i dati della funzione da visualizzare in valori compresi nel formato video. E cioè per il valore X = -3 pigreco della funzione il valore corrispondente da visualizzare sarà 1, e per il valore X = 3 pigreco deve essere 80.

Ma mentre lo scaling sulla X è controllato, poiché è sulla X che gira il loop principale, lo scaling della Y non è predeterminabile se non dopo aver calcolato tutti i valori Y da visualizzare. Nel nostro caso poiché la funzione definita in riga 110 non può superare i valori Y = -1.5 e Y = 1.5 si può prefissare in (-2, +2) il range della Y, senza pericolo di uscire fuori.

L'esercizio eseguito sul formato video alfanumerico nulla toglie all'efficacia dell'esercitazione.

Non abbiamo trattato, ma pensiamo di farlo prossimamente, il problema della tipologia delle scale. Poiché la scala è per definizione il rapporto numerico tra le grandezze misurate sull'output e quelle reali, esistono due differenti necessità:

— occorre conservare le proporzioni (il fattore di scala S è unico nelle due direzioni e le scale sono lineari)

— Non occorre oppure non si possono conservare le proporzioni (si possono usare Sx e Sy differenti, oppure i valori X e Y non sono omogenei (è il caso degli Istogrammi), oppure trattandosi di grafici di funzioni matematiche si usano scale non lineari (es. scale logaritmiche)).

Torneremo ancora sull'argomento. MC

Con Framework avrebbero fatto cose ancora piú grandi.

Leonardo (1452-1519)

Dante (1265-1321)

Galileo (1564-1642)

Cavour (1810-1861)

Dante, per esempio, avrebbe senz'altro sfruttato la procedura "profili" e i programmi di elaborazione testi di Framework. attraverso il video del suo Personal Computer avrebbe potuto buttare giu le prime idee (usando un riquadro per ognuna). E ritoccare lo scritto fino a dare forma completa ai sui capolavori.

Pensate: senza nemmeno un errore e in pochissimo tempo!

Oppure Leonardo avrebbe potuto utilizzare Framework per sviluppare la sua immaginazione creativa e per archiviare i soggetti dei suoi disegni.

Il suo genio, forse, sarebbe volato piu in alto.

E Galileo? Avrebbe potuto usare tutta la potenza di Framework per effettuare calcoli, collegamenti e prove senza perdere mai di vista le stelle.

D'altra parte, Framework sarebbe stato utile anche a Cavour. Per analizzare gli avvenimenti, per visualizzare con grafici le mosse degli avversari e per prevedere gli effetti di una decisione.

Avrebbe potuto realizzare anche ottime statistiche.

E oggi, a chi serve Framework? Praticamente a tutti. Anche a voi, perche e in grado di aiutarvi a risolvere tutti i problemi. E' un fantastico programma Ashton-Tate, capace di svolgere, da solo, una grande mole di lavoro: analisi, calcoli, previsioni, testi, comunicazioni, grafica, gestione dati...

Vi permette di ridurre i tempi delle decisioni e di aumentare i margini di sicurezza.

Framework. Un grande programma che tutti capiscono, perche parla italiano.

Framework e distribuito dall'Editrice Italiana Software.

ASHTON·TATE

I programmi della Ashton-Tate sono tradotti in Italiano e supportati dalla Editrice Italiana Software

Editrice Italiana Software S.p.A.

Editrice Italiana Software S.p.A.
Foro Bonaparte, 48 - 20121 - Milano
Tel. (02) 877312 - 877983

di Tommaso Pantuso

Un po' di logica

Un gran numero di lettori ci scrive chiedendo di saperne di più sul comportamento di tutte quelle "scatolette nere", i circuiti integrati, che si possono osservare aprendo un computer, per cominciare ad avere le idee più chiare sulle funzioni da esse svolte una volta collegata la presa di alimentazione. È un'esigenza che riteniamo giustificata vista la grande diffusione di queste macchine, e dell'elettronica digitale in genere, per cui non possiamo ignorarla. Più volte sulla nostra rivista (ed in particolare in questa rubrica) abbiamo affrontato alcuni problemi legati all'hardware —vedi ad esempio gli standard di comunicazione, le interfacce, le memorie, le tecniche di selezione dei dispositivi sul bus ecc.— dando per scontate alcune nozioni basilari a cui forse non tutti sono iniziati, trattate tra l'altro molto velocemente in tempi remoti in altre sezioni di MCmicrocomputer.

Analizzando tutti questi elementi, abbiamo deciso di cominciare a fornire al lettore, partendo da zero, tutte quelle nozioni che lo introdurranno gradualmente, e, speriamo senza molte difficoltà, nei concetti fondamentali dell'elettronica digitale. Non solo

Cercheremo, quanto più semplicemente possibile, di spiegare il funzionamento pratico di alcuni dei più diffusi circuiti integrati, da quelli "piccoli" a quelli "grandi", che serviranno a concretizzare gli argomenti trattati nel corso degli articoli proposti.

Prime nozioni

In elettronica digitale vengono messe in pratica le regole dell'algebra binaria definendo in maniera pratica alcune operazioni da essa teorizzate e facendole svolgere da appositi circuiti detti "circuiti logici".

Ciò che ci importa sapere per il momento è che l'algebra in questione basa la sua realtà su una variabile (binaria) che può assumere, in maniera mutuamente esclusiva, solo due valori — che possiamo chiamare 1 e 0 — con i quali è possibile stabilire se un certo evento si è verificato oppure no: se l'evento si è verificato diremo che esso è "vero", in caso contrario diremo che è "falso".

Naturalmente ci rendiamo conto che le affermazioni fatte finora sono alquanto sibilline, specialmente per i non iniziati all'argomento per cui, per spiegarci meglio, facciamo degli esempi.

Supponiamo di avere una scatola e voler verificare una delle seguenti due condizioni:

1) la scatola contiene degli oggetti
2) la scatola non contiene degli oggetti

È evidente che le due possibilità non possono essere vere contemporaneamente quindi, se è vera l'una, deve per forza essere falsa l'altra (mutua esclusione). Non possiamo infatti dire che la scatola è vuota se, ad esempio, vi mettiamo dentro una moneta e viceversa. "Scatola piena" o "scatola vuota" sono quindi le sole due possibili risposte al nostro interrogativo.

Lo stesso tipo di argomentazioni possono essere indotte, ad esempio, dall'osservazione della lampadina (una sola) della nostra stanza: essa può essere accesa o spenta e i due eventi non posssono verificarsi contemporaneamente per ovvi motivi. Se consideriamo i nostri occhi uno strumento valido di rilevazione dell'evento, potremo dire di vedere la luce emessa quando la lampadina è accesa e di non vederla quando è spenta.

Facciamo un passo avanti nel ragionamento verificando se un evento è vero o è falso a seconda che trovi riscontro o no in una definizione da noi formulata. La definizione è:

un uomo è alto se supera o al più è uguale ad un metro e settantacinque; è basso se è di altezza inferiore.

È evidente che considereremo allora alte le persone di un metro e settantacinque, di un metro e ottanta e così via mentre diremo che sono basse quelle di un metro e settantaquattro, di un metro e sessanta ecc. Le cose, come è facile osservare, vanno in maniera molto rigorosa, cioè non esistono vie di mezzo (un uomo di un millimetro inferiore a un metro e settantacinque viene considerato basso anche se è sulla soglia della condizione opposta) ed in questo fatto è insita la potenza dell'algebra binaria.

Livelli logici

Una condizione binaria di grande importanza è data dalla presenza o dall'assenza di una tensione in un punto (più precisamente tra un certo punto e la terra). Se infilate le dita in una delle prese si corrente di casa vostra, possono verificarsi due eventi: se la presa funziona (cioè se è allacciata al resto della rete elettrica), "prenderete la scossa" mentre se la presa non funziona, non rileverete su di essa nessuna corrente e quindi "non prenderete la

scossa". L'elettronica digitale, anche se in maniera meno "elettrizzante", basa tutte le sue elucubrazioni sul fatto che in un certo punto di un circuito elettrico "venga riscontrata" o "non venga riscontrata" una tensione. Diventiamo più seri per qualche istante e spieghiamoci meglio.

Cerchiamo, in termini pratici, per prima cosa di dare un significato alla parola "elemento digitale".

Un elemento digitale è un dispositivo sulla cui uscita è possibile rilevare, in un certo istante, solo uno di due possibili valori di tensione. Per fissare le idee supponiamo che tali valori siano rispettivamente 0 volt (assenza di tensione) e +5 volt (presenza di tensione). Possiamo quindi dire in altro modo che, su quell'uscita, o c'è una tensione o non c'è. A ciascuna delle due possibili condizioni assunte dall'uscita in questione diamo il nome di "stato" e diciamo, in maniera più sintetica, che l'uscita è nello stato "1" o a livello alto o, più semplicemente, a "1" quando su di essa è presente la tensione di 5 volt mentre diremo che la stessa uscita è nello stato "0" o a livello basso o a "0" quando su essa non è presente alcuna tensione (è presente la tensione di 0 volt).

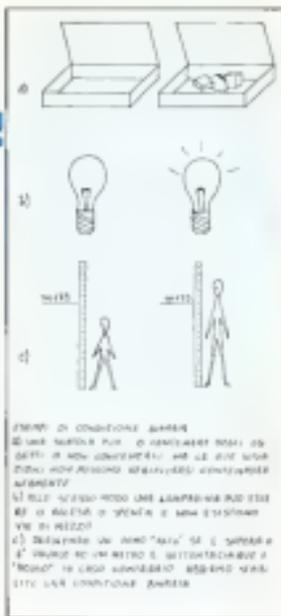
Anche se abbiamo a che fare con un dispositivo a più uscite, possiamo dire che esso è digitale se ciascuna di tali uscite rispecchia le condizioni testé illustrate.

Tabelle della verità

Una tabella della verità sintetizza un insieme di regole utili che stabilisce qual è l'evento risultante in seguito alla combinazione di altri eventi.

Un esempio di tabella della verità può essere il seguente:

evento 1	evento 2	risultato
piove	c'è vento	sono triste
non piove	c'è vento	sono triste
piove	non c'è vento	sono triste
non piove	non c'è vento	sono allegro

Non è difficile capire che è immediato conoscere il mio umore osservando come si combinano l'evento 1 e l'evento 2, cioè guardando fuori dalla finestra e verificando le due condizioni atmosferiche indicate. Tali condizioni atmosferiche sono la causa che modifica il mio umore a seconda di come sono combinate e possiamo considerarle "la sollecitazione che io ricevo a causa della quale viene prodotto un cambiamento del mio umore".

a)

b)

c)

m 1,75

ESEMPI DI CONDIZIONE BINARIA
a) UNA SCATOLA PUÒ O CONTENERE DEGLI OGGETTI O NON CONTENERLI MA LE DUE SITUAZIONI NON POSSONO VERIFICARSI CONTEMPORANEAMENTE
b) ALLO STESSO MODO UNA LAMPADINA PUÒ ESSERE O ACCESA O SPENTA E NON ESISTONO VIE DI MEZZO
c) DEFINENDO UN UOMO "ALTO" SE È SUPERA O È UGUALE AD UN METRO E SETTANTACINQUE E "BASSO" IN CASO CONTRARIO ABBIAMO STABILITO UNA CONDIZIONE BINARIA

Sintetizzando ancora le cose, possiamo schematizzare il tutto come un sistema che riceve, come variabili in ingresso, le condizioni del tempo mentre, lo stato dell'uscita, è rappresentato dall'umore.

L'esempio fatto, se descrive abbastanza efficacemente il concetto di tabella della verità, non calza a pennello alle variabili binarie perché le condizioni che compaiono nella tabella sono più di due. In altre parole, in una tabella della verità che descrive una certa operazione binaria dovremo trovare o solo le due condizioni "piove/non piove", o solo "c'è vento/non c'è vento", o solo "sono triste/sono allegro", resta naturalmente valido il concetto di ingressi e di uscita.

I VALORI DI TENSIONE SUI TERMINALI DI UN DISPOSITIVO DIGITALE TEORICAMENTE POSSONO ESSERE SOLO DUE NATURALMENTE, CONSIDERANDO UNA USCITA PER VOLTA, È IMMEDIATO CAPIRE CHE TALI VALORI NON POSSONO ESSERE ASSUNTI CONTEMPORANEAMENTE

Per chiarire meglio le idee su questo fatto, vediamo come può essere fatta una tabella della verità per una variabile binaria.

Supponiamo di essere in un stanza insieme ad altre due persone, ciascuna delle quali ci fornisca un'informazione, vera o falsa, in base alla quale noi possiamo dedurre se un altro evento, specificato a priori, è anch'esso vero o falso con le seguenti modalità: se le due informazioni ricevute sono entrambe vere, l'evento è vero mentre, se una sola delle due informazioni preliminari ricevute è falsa, allora l'evento in questione è falso (tra breve capirete meglio con un esempio). Se consideriamo le due informazioni ricevute come ingressi e la risposta ad esse come uscita, potremo comporre facilmente la seguente tabella della verità:

ingresso 1	ingresso 2	uscita
vero	vero	vero
vero	falso	falso
falso	vero	falso
falso	falso	falso

Vediamo come applicarla con qualche esempio.

Sia dato, come evento di cui bisogna stabilire la veridicità, la seguente affermazione:

Napoleone Bonaparte è vivo.

La prima informazione che riceviamo dai presenti è:

il cielo è celeste;

mentre la seconda è:

il triangolo ha dieci lati.

Analizziamole:

il cielo è celeste = vero

il triangolo ha dieci lati = falso

andando a confrontare la tabella della verità data, vediamo che all'associazione vero/falso corrisponde la risposta falso. L'evento proposto è quindi falso, cioè Napoleone "non" è vivo.

La tavola fornita può essere anche rappresentata, sempre con lo stesso significato, in maniera diversa. Se ad un evento vero associamo il simbolo "1" e ad uno falso il simbolo "0", se chiamiamo il primo ingresso x1 (prima informazione), il secondo x2 (seconda informazione) e l'uscita y,

la tabella proposta assume la seguente forma

x1	x2	y
1	1	1
1	0	0
0	1	0
0	0	0

che è molto più simile a una di quelle che potreste trovare in un libro che descrive le funzioni logiche di un circuito digitale. In conclusione di questo primo articolo, concretizziamo quanto detto finora con un'applicazione più concreta.

Una prima operazione logica

Prendiamo in considerazione ancora la tabella che è risultata dalle nostre divagazioni precedenti.

Neanche a farlo apposta (si fa per dire ..) essa definisce una ben determinata operazione logica. Descrive infatti il risultato dell'operazione di moltiplicazione binaria tra due elementi: il risultato è 1 solo se entrambi gli operandi sono diversi da zero.

Se indichiamo il simbolo di operazione binaria con una "x" tra un operando e l'altro, possiamo riscrivere la tabella precedente nella seguente maniera.

1 × 1 = 1
1 × 0 = 0
0 × 1 = 0
0 × 0 = 0

Fino ad ora abbiamo visto solo simboli. Vogliamo a questo punto mostrarvi, con un classico esempio, come alla tabellina ricavata sia possibile associare un ben determinato significato fisico.

Consideriamo un circuito alimentato da una batteria, con in serie due interruttori e una lampadina (confrontate una delle figure).

Se ad un interruttore aperto associamo il simbolo "0", ad uno chiuso il simbolo "1", alla lampadina, quando è accesa, il simbolo "1" e quando è spenta il simbolo "0", osservando il circuito non è difficile ricavare la tabella che segue:

interr 1	interr 2	lampadina
chiuso (1)	chiuso (1)	accesa (1)
chiuso (1)	aperto (0)	spenta (0)
aperto (0)	chiuso (1)	spenta (0)
aperto (0)	aperto (0)	spenta (0)

Confrontando l'associazione dei simboli (tra parentesi) effettuata come indicato in precedenza, non è difficile dedurre che essa equivale a quella della tabella precedente quindi, come si suol dire, ... i conti tornano.

Conclusioni

Quello di oggi è un primo approccio all'algebra della logica. Abbiamo visto come, partendo da un punto di vista del tutto generale, definendo una variabile binaria ed in seguito una certa operazione tra eventi, siamo giunti a dare ad un'associazione di simboli un significato fisico.

Esistono un gran numero di operazioni della stessa natura, ma con rapporto ingressi/uscita (o uscite) diversi, ricavate a partire da operazioni elementari. La prossima volta analizzeremo appunto tali operazioni e cercheremo di dar loro un significato fisico legandole a circuiti reali. Vedremo inoltre che esistono dei componenti, detti circuiti integrati, per mezzo dei quali è possibile ottenere elettricamente tali operazioni.

Le basi del
Data Base

Data Base Management System: sistemi per microcomputer

di Andrea de Prisco

Siamo giunti a una prima puntata conclusiva del nostro viaggio nei sistemi di gestione per basi di dati. Dopo aver parlato degli aspetti teorici dei modelli di dati generalmente usati, questo mese vedremo i Data Base dal punto di vista del lettore medio di MC un utente di personal computer. Cosa è possibile e cosa è impossibile (o quasi) implementare su un microcomputer.

Settima parte

Micro, home o personal?

È difficile stabilire con esattezza a quale di queste tre categorie appartiene un qualsiasi computer. O perlomeno sembra impossibile delimitare un netto confine tra le tre fasce. In virtù anche dei continui cambiamenti di mercato e, perchè no, di prezzo delle macchine.

Di macchine appartenenti alle tre categorie, è risaputo, ce n'è per tutte le tasche. Il Vic-20 è certamente un home, almeno a giudicare dal suo prezzo e/o dalle sue capacità.

L'M 20 della Olivetti è un personal, così come l'IBM Personal Computer.

Il Commodore 64, nato certamente come personal dalle capacità interessantissime (alla presentazione un paio d'anni fa, spacciato perfino come CP/M compatibile) sembra essere diventato il video game per eccellenza, a giudicare dall'età media dei suoi possessori e dal software disponibile, 90% giochi.

Che dire allora dei prezzi di questi mostriciattoli in continuo calo, in alcuni casi ribassati anche del 70%?

Una distinzione basata su valutazione a carattere prezzereccio, sembra essere dunque la meno attendibile. Ma allora, il nostro computer è un home, un personal o un micro?

E che dire delle periferiche? Esistono personal driver e home driver? Personal stampanti e home stampanti?

Qualcuno comincerà a chiedersi cosa c'entra tutto questo discorso con i data base, argomento di queste puntate. La chiave sta tutto nel sottotitolo: sistemi per micro computer.

Cioè?

Scopriremo che esistono Home Data Base, Personal Data Base e Micro Data Base.

Home Data Base

...o per meglio dire, Data Base D.O.C. (d'origine casereccia).

Secondo una recente definizione, una Base di Dati è un insieme di dati strutturati e permanenti, raggruppati in insiemi omogenei in relazione fra loro, organizzati con la minima ridondanza per essere usati da applicazioni diverse, in maniera controllata <Albano 84>.

A tutt'oggi, il termine data base, di contro, è continuamente nominato (...e blasfemato) per indicare perfino semplici programmini Basic che permettono una mini-gestione di un mini-indirizzario su mini-videogiochi che tanto stanno facendo impazzire (a causa della nota game-dipendenza) ragazzetti di tutte le taglie.

Parlare di data base vuol dire fornire specifici strumenti per la definizione e il trattamento di grandi moli di dati, strumenti per assicurare l'integrità e la sicurezza degli stessi, nonché un vero e proprio linguaggio di programmazione usato per scrivere proprie procedure e utility che sfruttano la base di dati installata. Il produttore medio di software questo non lo vuole capire e spesso produce (secondo lui) data base che non sono degni nemmeno di essere chiamati sistemi di archiviazione.

L'opportunità di organizzare la base di dati in classi di elementi, e non in un unico insieme dalle dimensioni enormi, nasce essenzialmente da due necessità: minima ridondanza, duplicazione di informazione e massimo sfruttamento di quest'ultima. Anche se ne abbiamo già ampiamente parlato 5 numeri fa, per intenderci su questo punto, facciamo un esempio: vogliamo gestire i dati della segreteria studenti di una università, e in particolare gli studenti iscritti a un corso di laurea e gli esami dell'ordinamento dello stesso.

La situazione è mostrata in figura 1: abbiamo due classi, Studenti ed Esami, e l'associazione multipla tra questi. Multipla sta nel fatto (ovvio) che ogni studente può aver superato più di un esame e che ogni esame del corso di laurea è stato superato da più studenti. Avremo cioè che ogni elemento della classe Studenti sarà associato con uno o più elementi della classe Esami e viceversa.

Indipendentemente dal modello di dato usato (semantico, gerarchico, reticolare o relazionale) e a eventuali trucchetti, discussi nei numeri scorsi, per implementare tale situazione quando non direttamente possibile, la scomposizione in due classi di oggetti tra loro correlati permette ad esempio di conoscere (subito, facilmente) quali esami ha sostenuto un determinato studente, o quali studenti hanno superato un determinato esame, partendo (sempre ad esempio) dall'indirizzo di uno studente o dal nome di battesimo del professore che tiene un corso (e che presiede alla sessione d'esame).

Per di più, modificare la base di dati vista sopra, si riduce a informare il sistema che "Lo studente Tizio ha superato l'esame BlaBlaBla".

Personal Data Base

Non sono ancora Data Base effettivi, ma ci manca poco. A questa categoria appartengono potentissimi sistemi di archiviazione che permettono in alcuni casi perfino di programmare, con un opportuno linguaggio fornito per il trattamento dei dati, le applicazioni necessarie all'utente (generazione di rapporti, statistiche sui dati, ecc.).

L'unica vera limitazione sta nella mancanza del meccanismo delle classi che, come visto sopra, permette di modellare mol-

to facilmente aspetti assai comuni nella realtà

In un sistema di archiviazione, per modellare la situazione studenti-esami si può procedere in due distinti modi: utilizzare un archivio unico o due archivi non (implicitamente) in relazione tra loro

Dato che per ogni archivio che si definisce è possibile specificare il nome dei vari campi delle registrazioni, il primo modo consiste nell'inserire l'elenco degli esami sostenuti come proprietà del singolo Studente. Ogni elemento di questo super archivio, avrà come campi: Nome, Cognome, Matricola, Indirizzo, Telefono, DataDiNascita, Esame1, Esame2, ... EsameN (se N sono gli esami del corso di laurea in questione)

Al momento dell'iscrizione un qualsiasi studente avrà tutti gli esami "vuoti". Man mano che supererà esami si dovranno riempire le varie colonne della registrazione riguardante lo studente in considerazione

È ovvio che se 100 studenti hanno superato l'esame "Letteratura Italiana dal 1600 al 1700", tale stringa sarà necessariamente presente cento volte nell'archivio in cento posti diversi

Se poi per ogni esame ci interessa anche il nome e il numero telefonico del professore che presiede alla sessione d'esame (il titolare del corso) la situazione, quanto a spreco di spazio, comincia a farsi preoccupante. Dovremo infatti inserire anche Nome1 ... NomeN e Telefono1 ... TelefonoN come proprietà di ogni Studente. Posto che il professore dell'esame di sopra si chiami "Massimiliano Mazzanti Viendalmare" e abbia come numero di telefono 353637383 anche tutta quest'altra robaccia sarà presente cento volte nell'archivio

È chiaro che tale situazione non salterà mai in mente a nessuno di implementarla così com'è. Meglio ricorrere a due archivi separati, uno contenente informazioni riguardanti gli studenti, l'altro gli esami di un corso di laurea. Si cerca cioè di spingersi maggiormente verso una specie (permetteteci questo termine) di DataBasizzazione dell'archivio, purtroppo con notevoli limitazioni rispetto ai Data Base veri

Inseriremo come proprietà degli studenti, oltre alle informazioni riguardanti nome, matricola, eccetera, lo spazio per i soli codici degli esami sostenuti (il codice identifica univocamente un esame)

L'archivio Esami avrà i seguenti campi: Codice, NomeEsame, NomeProfessore, TelefonoProfessore. Per ogni studente che supererà un esame, bisognerà accedere all'archivio studenti e inserire il nuovo codice nella registrazione riguardante lo studente

L'affare si complica un po' se vogliamo conoscere il nome di tutti i professori che hanno "esaminato" lo studente Tizio. Per prima cosa si apre l'archivio studenti (a mo' di semplici file su disco, generalmente ogni archivio prima di essere usato va aperto). Si accede alla registrazione con una semplice interrogazione al sistema: in quanto a queste nulla da recriminare, i sistemi di archiviazione sono particolarmente potenti in merito a selezione di elementi o ordinamenti di archivi secondo qualsiasi criterio

Una volta selezionato lo studente grazie alla Biro-device (penna a sfera) si trasferiscono su supporto cartaceo (block-notes, Scottex casa o in mancanza d'altro anche il tavolo del computer) i codici degli esami da lui sostenuti. Prima di aprire l'archivio Esami si chiude l'archivio Studenti (non sempre è possibile mantenere aperti più archivi). Non resta che chiedere al sistema tutti gli esami che hanno come codice un codice trovato precedentemente presso lo studente Tizio e leggere direttamente su monitor i nomi cercati

Se il sistema dispone di un proprio linguaggio di programmazione dotato di tutti gli operatori per il trattamento dei dati in memoria di massa, è possibile automatizzare questa procedura (chiamandola ad esempio EsaminatoriDi) rendendo il tutto ancora più vicino ai Data Base veri e propri

Micro Data Base

A questo punto, se non avete a disposizione un IBM PC o un M 20, con tanto di disco rigido da 10 Mega in linea, scordatevi pure di veder realizzare queste cose sul vostro personal. Non siamo cattivi: è solo che per fare davvero le cose per bene, è ridicolo pensare a un Commodore 64 con tanto di 1541 appiccicato

Comunque, per non lasciare a boccasciutta l'utente medio, nel prossimo numero vedremo cosa sia possibile fare con un 64: presenteremo un data base che, seppur molto limitato in quanto a interrogazione e a quantità di dati trattabili, perlomeno avrà la possibilità di gestire i dati per mezzo di classi e (udite, udite!) con correlazione tra le registrazioni presenti in memoria di massa

I Micro Data Base non sono altro che versioni leggermente semplificate dei Sistemi di gestione per Basi di Dati propri dei grossi calcolatori IBM, Honeywell, HP e consimili

Le limitazioni riguardano essenzialmente il numero massimo di utenti che possono accedere al sistema contemporaneamente da più terminali (mediamente 8, ma in alcuni casi si può arrivare anche a 127) e dalla mole di dati trattabili, dovuta alla quantità di memoria (centrale e di massa) di cui si dispone

Per quanto riguarda i sistemi monoutente (o personali se preferite) la tendenza principale è di fornire un prodotto quanto più facile da usare e con una flessibilità tale da permettere all'utente stesso di apportare delle modifiche all'organizzazione della Base di Dati adattandola alle sue mutevoli esigenze

Per il resto, i Micro Data Base, sono strumenti potentissimi che permettono di trattare i dati allo stesso modo dei grossi calcolatori. Avremo MBD reticolari, relazionali con tutti gli operatori propri del modello, nonché potentissimi strumenti di programmazione per poter automatizzare le procedure che si desiderano

Anche l'integrità dei dati e la sicurezza di questi è garantita dagli stessi meccanismi

LOG... contro il mal di black-out

Qui si parla di togliere e inserire elementi in una base di dati come fossero "bruscoletti". Non s'è detto, sebbene l'utente possa ignorarlo del tutto, che ogni registrazione in una base di dati non è ... solo registrata.

Ad ogni inserimento bisogna aggiornare una marea di puntatori, avvertire gli elementi padre che è nato un altro figlio, tutti gli indici su chiavi secondarie devono essere aggiornati. Solo così sono possibili ricerche veloci al momento opportuno. Idem dicasi per l'operazione di cancellamento: bisogna fare macchina indietro in ogni direzione

Non a caso molti sistemi di gestione di base di dati, dopo una richiesta di modifica e prima del nuovo Prompt rispondono "Commit" (trad. commesso, nel senso di commettere ... lascia intendere qualcosa di molto scottante ...)

E se manca la corrente nel bel mezzo di una modifica? Magari può anche verificarsi che l'elemento è stato tolto, ma se tutti i puntatori non sono aggiornati, la base di dati passa in uno stato inconsistente. Può darsi ad esempio che un elemento non sia più presente in una classe, ma risulta ancora essere figlio di un altro elemento in un'altra classe. O viceversa

Dipende da quale operazione il sistema fa prima e quale dopo, sempre serialmente. In ogni caso è un bel pasticcio, anche perché spesso non si può cancellare un elemento solo dagli indici o solo da una classe

Per ovviare a questo genere di inconvenienti (già, non sempre la colpa è dell'Enel: potrebbe disintegrarsi una unità a dischi per colpa dei raggi cosmici ...) è buona norma fare delle copie periodiche di tutta la base di dati su altro supporto, magari conservandolo in ambienti diversi dalla Sala Macchine

Solo questo, è troppo banale: l'elemento essenziale è il giornale delle modifiche, o LOG, che è un apposito file contenente la lista di tutte le operazioni effettuate con successo dal momento dell'ultima copia della base di dati sino al giorno disgraziato

Quando si esegue una modifica, questa viene prima inserita sul giornale, poi eseguita e se tutto è a posto, una apposita marca sempre sul LOG, indica che ha ottenuto successo e che quindi ha cambiato lo stato della base di dati (rimanendo però consistente)

Se avviene qualche inconveniente per il quale la base di dati passa in uno stato inconsistente, niente paura: si prende l'ultima copia della base di dati di cui si dispone e a partire da questa il sistema esegue una per una tutte le modifiche (con successo) riportate sul giornale. La base di dati si ritroverà nello stesso stato dell'ultima operazione eseguita con successo prima del Black-out: non resta che ripetere il comando di modifica, sperando che non succeda dell'altro

dei sistemi più grossi, quali l'aggiornamento automatico del giornale delle modifiche e la possibilità di fare delle copie periodiche di tutta la base. Con questi meccanismi si può stare al sicuro praticamente da tutti gli incidenti compreso il Black-Out nel bel mezzo di una transazione (a causa di una modifica, la base di dati passa da uno stato a un altro). Nell'apposito riquadro a pag 90 è mostrato come è realizzabile tale sicurezza

Vedremo ora alcuni Data Base per microcomputer iniziando dal basso col Superbase 64 adatto all'ultradiffuso Commodore

Superbase 64

Delle tre categorie sopra discusse, certamente questo appartiene alla categoria dei personal Data Base, essendo, di fatto, più simile a un potentissimo sistema di archiviazione. Niente classi o correlazioni tra dati, la potenza del Superbase 64 è tutta nella facilità d'uso e nelle molteplici possibilità di interrogazione

Questo "quasi" data base è di tipo a maschera le registrazioni hanno un proprio formato di uscita su video, dichiarato assieme al tipo dei campi al momento della creazione di un file. Si ha a disposizione l'intero schermo per poter "disegnare" la forma della generica registrazione di ogni file creato. Per esempio, se si sta definendo un archivio indirizzi vogliamo il nome nella seconda linea dello schermo, immediatamente sotto il cognome, di seguito (o qualche linea più giù) vogliamo il campo indirizzo con a fianco la città e così via fino a completo riempimento di tutto lo schermo. Anzi, se non dovesse bastare, è possibile richiederne altri per inserire ulteriori campi nella dichiarazione del nostro record

Oltre a questo, nelle varie schermate possono essere aggiunti altri caratteri grafici per abbellire un po' il contesto

Definita la maschera delle nostre registrazioni possiamo procedere all'inserimento dei dati. Il sistema visualizza la schermata col cursore lampeggiante nel primo campo. È possibile anche muoversi nei vari campi, semplicemente usando i tasti di controllo cursore. Per la ricerca di elementi si procede in modo analogo il sistema visualizza la schermata con i vari campi vuoti, e dopo averne riempito qualcuno per permettere al computer di individuare la registrazione si preme [Return]. Ad esempio, vogliamo trovare la prima registrazione che ha come campo Città

I modelli di dato

Faremo un piccolo riassunto dei modelli di dato visti negli ultimi 4 numeri, con particolare riferimento al modo di concepire l'associazione tra elementi di classi diverse. Il primo modello analizzato è quello semantico: il più innovativo di tutti essendo anche l'ultimo nato (in senso buono)

Modellare particolari situazioni col modello semantico dei dati è abbastanza semplice: abbiamo ad esempio l'associazione Studenti-Esami di figura 1. Ogni studente ha superato più esami, ogni esame è stato superato da più studenti associazione multipla. In linguaggio letterale, la registrazione riguardante uno Studente avrà un campo Nome, un campo Cognome, un campo Matricola, e una sequenza di Esami (quelli da lui superati). Un Esame avrà un campo Nome, un campo Codice e la sequenza di Studenti che l'hanno superato. Tradotto in un linguaggio di tipo semantico scriveremo qualcosa del tipo

```
class Studenti <-> (Nome:string and
Cognome:string and matricola int and esami Superati: seq Esami)
e
class Esami <-> (Nome:string and Codice: string and SuperatoDa: seq Studenti)
```

Il modello dei dati gerarchico, essendo il più anziano è anche quello più vincolato. Esiste un solo tipo di associazione, con diretta multipla e inversa univoca e ogni classe può essere al più codominio di un'altra classe. In figura 2 è mostrato un esempio rappresentabile col modello gerarchico. Un dipartimento ha più docenti e più studenti iscritti, ma tanto i primi quanto i secondi non possono appartenere a più di un dipartimento

In figura 3 è mostrato un esempio reticolare. In questo modello ogni classe può essere codominio di più classi, ma il vincolo dell'univocità dell'inversa resta. Ogni Studente appartiene a un solo Dipartimento e ogni Dipartimento ha più Studenti iscritti. Idem dicasi per l'associazione Dipartimenti-Docenti. La terza associazione (univoca) tra Corsi e Docenti lega un corso al suo Docente

I data base Relazionali devono il loro nome al concetto di relazione (tabella di elementi, vedi fig. 4). Il meccanismo che permette associazione tra i dati è detto delle chiavi esterne, e si avvale di un particolare operatore relazionale detto JOIN. Il JOIN di due tabelle ne costruisce una terza, accoppiando a ogni elemento della prima tutti gli elementi della seconda che soddisfano una particolare condizione data assieme al comando. Per fare un esempio, immaginiamo che la relazione di figura 4 rappresenti nome e telefono dei clienti di una ditta. Supponiamo di avere un'altra tabella di tutti i clienti che non hanno ancora saldato un conto avremo per ogni elemento un campo NomeCognome e un campo ImportoDovuto. Se vogliamo telefonare per un sollecito a tutti i clienti morosi faremo il join tra la tabella crediti e la tabella utenti con condizione di uguale NomeCognome. Otterremo in questo modo una tabella con elementi formati da tre campi NomeCognome, ImportoDovuto e Telefono

Dito indice alla mano, possiamo alzare la cornetta telefonica

```
+----------+                            +----------+
!          ! HaSuperato    SuperatoDa   !          !
! Studenti !---------->><>〈<-----------!  Esami   !
!          !                            !          !
+----------+                            +----------+
```

Figura 1

```
          +-----------+
          !           !
          !Dipartimenti!
          !           !
          +-----------+
           /         \
+----------+         +----------+
!          !         !          !
! Docenti  !         ! Studenti !
!          !         !          !
+----------+         +----------+
```

Figura 2

```
+----------+          +-------------+
!          !          !             !
! Studenti !<---------!Dipartimenti !
!          !          !             !
+----------+          +-------------+
                             |
                             v
+----------+          +-------------+
!          !          !             !
!  Corsi   !--------->!   Docenti   !
!          !          !             !
+----------+          +-------------+
```

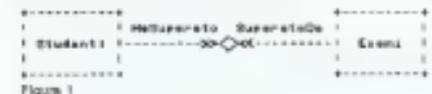
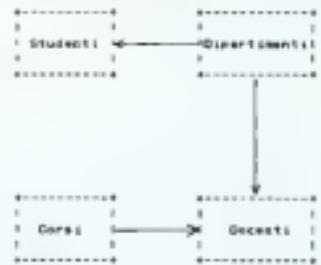



Figura 3

NomeCognome	Telefono
Mario Rossi	877877
Gino Verdi	765654
Luca Bianchi	546578
Nino Neri	865584

Figura 4

"Pisa". È sufficiente digitare Pisa nel corrispondente campo della schermata e battere [Return]. Il sistema si posizionerà sulla prima registrazione che soddisfa tale condizione (Città = Pisa). Da questa posizione del nostro file è possibile chiedere il successivo, il precedente o tornare in un sol colpo al primo elemento del file.

Per quanto riguarda i tipi di campi ammissibili, dobbiamo dire che anche qui la Precision Software (madre sia del Superbase 64 che del famosissimo Easy Script) non ha badato a spese. Sono infatti disponibili campi di tipo testo (lettere, numeri e simboli), numerici interi e decimali, campi di tipo Data (con calcolo automatico del corrispondente giorno della settimana) e di tipo Result ossia risultato di operazioni con altri campi, a mo' di tabellone elettronico. Per ogni registrazione è obbligatorio definire un campo chiave tramite il quale è possibile un accesso molto veloce alla singola registrazione. Per finire la possibilità di scrivere propri programmini in un linguaggio simile al Basic, naturalmente fornito di tutti gli operatori per accedere alle registrazioni dell'archivio.

In parole tecniche (ne abbiamo parlato nel numero scorso) il Data Manipulation Language del Superbase 64 è ospite del Basic standard. È chiaro che quanto detto in queste righe è solo una lieve infarinatura delle varie possibilità offerte dal Superbase 64. Del resto non è questa la sede più opportuna per spiegarne appieno tutte le feature, sperando di poter presto ritornare sullo specifico tema con una prova su strada di questo particolare package.

dBASE II

Nato come sistema di gestione di base di dati per microcomputer dotati di sistema operativo CP/M, il dBASE II è stato successivamente adattato anche per l'uso sotto MS-DOS (microprocessori a 16 bit). Relazionale monoutente, questo sistema dispone di molte caratteristiche interessanti: si tratta certamente di un package destinato a un uso professionale.

Per iniziare diciamo subito che il dBASE II è un data base vero, e non (tanto per cambiare) un ennesimo sistema di archiviazione evoluto. Spiccano tra le sue caratteristiche principali un buon linguaggio di programmazione a se stante (dotato cioè di proprie strutture per il controllo del flusso), l'indipendenza fisica e logica tra organizzazione dei dati e programmi, e non ultima la possibilità di leggere archivi generati da altri linguaggi di programmazione quali il FORTRAN o il PASCAL.

Queste due ultime peculiarità fanno sì che eventuali (piccole) modifiche alla struttura dei dati non si ripercuotano sui programmi applicativi già scritti e che, passando (come spesso accade) da un sistema di archiviazione a un data base, è possibile leggere i vecchi file per caricare in una maniera più o meno automatica la base di dati che si vuole costruire.

Fra le limitazioni di questo data base (imposte essenzialmente dalle caratteristiche hardware delle macchine alle quali è destinato) annoveriamo la lentezza dei programmi applicativi, eseguiti caricandoli linea per linea dal disco e un non troppo flessibile data definition language: ogni registrazione può essere lunga al più un K e ogni campo del record al più 356 byte.

Essendo di tipo relazionale, la definizione della base di dati si ridurrà alla definizione delle varie tabelle (relazioni) da usare. Oltre a ciò, se interessa una maggiore velocità di ricerca per determinati campi di una tabella è possibile costruire automaticamente degli indici. Eventuali inserimenti o correzioni alla base di dati si ripercuotono implicitamente sulla ristrutturazione degli indici interessati.

Le operazioni possibili sui dati si riferiscono essenzialmente all'aggiunta di nuovi elementi in una tabella, al cancellamento (logico o fisico) di parte dei dati, e all'aggiornamento di record già esistenti. Oltre a queste è possibile una "totalizzazione" di alcuni campi ed eseguire l'operazione di giunzione (JOIN) tipica di ogni sistema relazionale.

La totalizzazione consiste nel fare la somma di alcuni campi numerici di tutti i record che hanno in comune altri campi. Facciamo un esempio, abbiamo la seguente tabella

Articolo	Quantità
VIC 20	100
SINCLAIR SPECTRUM	120
CBM 64	90
APPLE 2/C	105
CBM 64	35
VIC 20	120
SINCLAIR SPECTRUM	120
VIC 20	100

riguardante le ordinazioni pervenute ad un distributore di personal computer. Totalizzando la tabella sui campi Articolo otterremo

Articolo	Quantità
VIC 20	320
SINCLAIR SPECTRUM	240
CBM 64	125
APPLE 2/C	105

che corrisponde alla tabella degli articoli e dei totali delle quantità ordinate.

L'operazione di JOIN permette di creare tabelle a partire da due tabelle già esistenti: si usa per effettuare la famosa correlazione tra dati col meccanismo delle chiavi esterne, come già ampiamente discusso lo scorso mese e mostrato in sunto nel riquadro a pag. 91 di questo articolo.

MDBS

La sua sigla sta per Micro Data Base System e, attualmente, si può considerare il più potente tra i data base per microcomputer. Seguendo il modello di dati reticolare, pur non attenendosi strettamente alle specifiche standard CODASYL, offre tutte le potenzialità dei sistemi più grossi, e per alcuni versi anche qualcosa in più. Ad esempio la possibilità di definire le associazioni multiple tra dati appartenenti a insiemi diversi.

Possiamo dividere l'MDBS in 5 moduli:

1) DDL (data definition language)
2) DMS (data management system)
3) DRS (dynamic restructuring system)
4) RTL (recovery/transaction logging system)
5) QRS (query system/report writer)

Il DDL serve per definire la base di dati che si vuole installare. Lo schema generale di una definizione è il seguente:

<nome della base di dati>
<drive da utilizzare>
<diritti di accesso>
<struttura dei vari insiemi>
<struttura delle associazioni>

I diritti di accesso servono per proteggere dati da accessi indiscreti. È possibile ad esempio definire una PASSWORD senza la quale nessuno è autorizzato a curiosare nella BD o solo a fare modifiche.

Per ogni insieme si definisce a sua volta la struttura interna (nome e tipo di ogni campo della registrazione) e per le associazioni, automatiche o manuali, si deve indicare l'insieme Padre, l'insieme Figlio e il tipo di connessione (biunivoca, univoca-multipla, multipla-univoca o multipla).

Il DML, col meccanismo delle sotto routine, permette di operare sulla base di dati. La struttura generale di un comando assume la seguente forma:

EO = CALL(Indirizzo "ComandoVeroeProprio" ArgomentiLinguaggioOspite)

EO è una qualsiasi variabile che al ritorno dalla sottoroutine conterrà un valore relativo all'esito dell'operazione. Indirizzo è l'indirizzo di partenza della routine. Gli ArgomentiLinguaggiOspite servono per interfacciare il programma in esecuzione con DML.

Il modulo DRS permette di modificare dinamicamente la struttura della base di dati. È possibile ad esempio aggiungere nuovi insiemi, aggiungere campi a insiemi già esistenti, modificare associazioni, togliere insiemi. Tutto sempre senza dover reinserire dati in memoria.

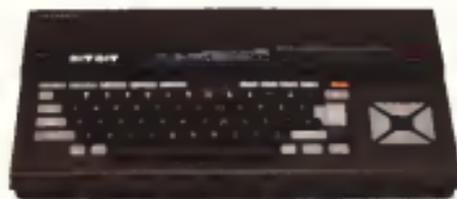
Se nel corso di una operazione dovesse verificarsi un errore hardware (compresa l'improvvisa mancanza di corrente) il modulo RTL provvede a riportare la base di dati in uno stato consistente, grazie all'utilizzo di un giornale delle modifiche.

Per finire, tramite il modulo QRS è possibile accedere alla base di dati in modo interattivo, semplicemente digitando comandi di ricerca o di modifica, nonché generare rapporti o costruire nuovi insiemi con registrazioni che soddisfano particolari condizioni. MC

Sony è lieta di presentare il primo studente che ha risolto i suoi problemi col computer.

Ormai gli emici lo chiamano Einstein. Da quendo ha l'Hit-Bit Sony, Andrea non he più paura di nessun problema, né di algebre né di geometria!

HOME COMPUTER HIT BIT

il nuovo computar Hit-Bit Sony è veramente facile. Quasi come scrivere a mecchina. Hit-Bit Sony è un vero computer "femillere", adatto per tutta la famiglie. La mamma lo usa par la dieta e per la dispensa di casa, papa per i conti del bilencio, per le denuncia dei radditi e per ie sua "collezione" di vini. Barbare per gli oroscopi, per i bioritmi e per tenere in ordina i dischi. Andrea per studiare (ci sono programmi di itellano, matematica, geometria, storia, geografia, ecc), per suonere le sue cenzoni e per un sacco di videogiochi. Insomma, con Hit-Bit Sony in poche settimane una normalissime femiglia si è trasformata in un'autentica "famiglia el computer".

DATA BANK PERSONALE. Una caratteristica che colloca Hit-Bit Sony al di sopra degli altri computer è il "Data Bank Personale", un programma incorporato che consente di organizzare con estrema facilità appuntamenti, promemoria, indirizzi e numeri telefonici, con la possibilità di immagazzinare 4 Kbytes di informazioni su cassette o sull'esclusive DATA CARTRIDGE HBI 55, con batterie incorporate contro le cancellazioni accidentali.

MSX* è la sigle del nuovo standard internazionale unificato, edottato dalle più importanti marche del mondo di Home Computer (Sony in testa). La caratteristica rivoluzionarie dell'MSX è la compatibilità: per la prima volta nella storia degli home computer, tante marche diverse parlano la stessa lingue, rendendo così possibile l'interscambio dei programmi e delle unità periferiche, (più o meno, quello che già succede coi componenti Hi-Fi)

(*) MSX è un marchio registrato della Microsoft Co

Sony HB-75 P	Scheda Tecnica
CPU	Compatibile Z80A
Memoria	ROM 32 Kbytes (BASIC) + 16 Kbytes (FIRMWARE) RAM 64 Kbytes + video 16 Kbytes
Schermo	Testi: 37 colonne da 24 linee (fino a 40 col.) Grafica: 256 x 192 segni. 16 colori.
Suono	Gamma ad 8 ottave, 3 generatori di tono
CMT	1200/2400 baud (FSK format)
Interfacce incorporate	CRT: RGB video e audio - RF (UHF 36 ch) - Stampante: CENTRONICS 8-bit - interfaccia parallela
Ingressi	Cartucce MSX x 2 - joystick x 2
Dimensioni e peso	mm 405 x 87 x 248 - Kg. 2,84
Unità periferiche	Plotter stampante a colori - Joystick - Joystick senza filo - Micro Floppy Disk Drive - Micro Floppy Disk - Data Corder

Hit-Bit Sony, il primo computer "familiare".

SONY.

SPECIALE
MSX
COMPUTER MAGAZINE
N 1/1985
Sped in abb post Gr III L 9 000

La Grafica (parte II)

Il mese scorso abbiamo visto quali sono le principali caratteristiche dei computer MSX dal punto di vista grafica, esaminando tra l'altro i due versatili comandi LINE e CIRCLE, che consentono di disegnare linee, rettangoli, archi, cerchi ed ellissi

L'argomento di questa seconda puntata sulla grafica MSX è il Graphics Macro Language (GML), un macro linguaggio per il tracciamento di linee

Il GML è composto da una serie di comandi che fanno muovere sullo schermo una penna immaginaria. Questa penna, che si può spostare soltanto in linea retta, muovendosi lascia un segno, in altre parole disegna un segmento. Combinando fra loro più istruzioni del GML è possibile disegnare figure, anche piuttosto complesse, molto più semplicemente che non attraverso una serie di LINE

I comandi del GML non possono essere eseguiti direttamente come le normali istruzioni del Basic, ma vanno prima inseriti in una stringa e poi mandati in esecuzione dal comando DRAW. In altre parole, per eseguire una serie di comandi del macro linguaggio grafico bisogna dare l'istruzione

```
DRAW "<saquenza di comandi GML>"
```

A questo punto possiamo iniziare a vedere quali sono e come sono fatti questi comandi, partendo dai più semplici

Per disegnare una linea in alto, in basso, a destra o a sinistra si usano le seguenti istruzioni

```
U <n> (alto)
D <n> (basso)
L <n> (sinistra)
R <n> (destra)
```

dove n è il numero che indica la lunghezza del segmento

Tutti i comandi del GML hanno questa forma, cioè una lettera maiuscola eventualmente seguita da uno o due numeri

Vediamo subito un primo esempio, per chiarire le idee supponiamo di voler disegnare un quadrato di 55 pixel per lato. La sequenza di comandi necessaria è

```
R55D55L55U55
```

e può essere eseguita con un'istruzione DRAW in questo modo

```
DRAW "R55D55L55U55"
```

Non abbiamo detto nulla però a proposito della posizione alla quale verrà disegnato il quadrato. La regola è questa: il disegno incomincia dalla posizione dell'ultimo pixel acceso, se ancora non è stato acceso nessun puntino si assume come default quello di coordinate 0,0

Il colore della figura sarà quello del corrente colore di primo piano

Un programma completo per disegnare il quadrato voluto è riportato nel listato 1, la parte fondamentale è costituita dall'istruzione DRAW appena vista, preceduta dalla chiamata allo schermo grafico numero 2 ed da un PSET (100,68) che fissa la posizione del quadrato

Per disegnare in diagonale si usano invece questi altri comandi

```
E <n> (in alto a destra)
F <n> (in basso a destra)
G <n> (in basso a sinistra)
H <n> (in alto a sinistra)
```

con queste nuove istruzioni possiamo provare a disegnare un rombo, il programma relativo è riportato nel listato 2

La struttura è la stessa del programma precedente: un'istruzione DRAW preceduta da uno SCREEN e da un PSET.

Dai due esempi visti finora si può comprendere che il principale vantaggio del GML rispetto all'istruzione LINE consiste in una maggiore compattezza nonché in una più facile comprensione dei comandi, che invece di fare riferimento a delle coordinate di schermo indicano semplicemente direzione e lunghezza della linea da tracciare

Le otto istruzioni che abbiamo presentato ci permettono tuttavia di disegnare solo in alcune direzioni ben definite, per tracciare un segmento in una direzione qualsiasi è

Output del programma Quadrato

necessario ricorrere al comando M, che ha questa forma

```
M <x>,<y>
```

x e y sono le coordinate del secondo estremo del segmento, come coordinate del primo estremo si assumono, al solito, quelle dell'ultimo pixel acceso.

In questo caso, come si vede, è necessario specificare delle coordinate e si viene a perdere uno dei vantaggi sopra accennati, rimane comunque quello della compattezza

Possiamo provare a modificare il programma QUADRATD per tracciare l'ultimo segmento con un M anziché con un U, la linea da cambiare naturalmente è la 90.

```
90 DRAW "R55 D55L55M100,68"
```

Premettendo alle coordinate x e y il segno + o —, esse vengono interpretate non come coordinate assolute, ma relative; provate ad esempio a modificare la linea 90 così

```
90 DRAW "R55D55L55M + 100, + 68"
```

il risultato che si ottiene è decisamente diverso!

Per avere il solito quadrato bisogna invece cambiare la stessa linea in

```
90 DRAW "R55D55L55M + 0 + 55"
```

Se il concetto di coordinate relative non vi è molto chiaro vi consigliamo di rileggervi la rubrica del numero precedente, al punto in cui si parla dell'istruzione PSET STEP

Immaginiamo ora di voler disegnare non una sola ma due figure diverse, ad esempio il quadrato e il rombo già visti

Un primo modo è questo

```
PSET (X1,Y1)
DRAW "<comandi disegno quadrato>"
PSET (X2,Y2)
DRAW "<comandi disegno rombo>"
```

i due PSET sono necessari per posizionare opportunamente le due figure

Effettivamente così funziona (provare per credere), ma esiste una strada ancora più semplice, che permette di usare un solo DRAW e di eliminare i PSET

Per capire come si fa bisogna prima parlare del comando B: premettendo questa lettera ad un altro comando l'ipotetica penna menzionata in apertura si sposta *senza* disegnare

Dato che le linee vengono tracciate sempre a partire dall'ultima posizione della penna (posizione che può essere stata rag-

Output del programma Rombo

giunta con un comando del GML oppure "forzata" da un PSET) l'effetto che si ottiene, in pratica, è quello di spostare il punto di inizio del disegno successivo

Il programma del listato 3 è un parziale esempio di quanto detto: esso traccia un quadrato ed un rombo scegliendo la posizione del rombo non con un PSET (120,91) bensì con un BM120,91

Anche il primo PSET poteva essere benissimo sostituito da un BM40,65, abbiamo preferito lasciarlo per rendere più visibile il punto di partenza generale. Sempre per motivi di chiarezza abbiamo utilizzato tre DRAW invece di una sola: è perfettamente lecito riunire tutti i comandi (e magari anche il PSET della linea 90, trasformato in un BM) in un'unica stringa, con un notevole risparmio di spazio che però sarebbe accompagnato da una perdita di leggibilità

Esiste anche un altro comando che permette di spostare la penna senza che essa disegni, ed è il comando N

Quando viene premesso ad un normale comando di tracciamento, N fa sì che la penna, una volta disegnato il segmento, torni al punto di partenza. Ecco il rituale esempio

```
10 SCREEN 2
20 PSET (128 96)
30 DRAW "NU20ND20NR20NL20
40 DRAW "NE15NF15NG15NH15"
50 GOTO 50
```

Questo programma disegna una specie di stella; per rendervi meglio conto dell'effetto di N potete provare ad eliminarlo completamente sia dalla linea 30 che dalla 40 per vedere cosa cambia

Notate che anche in questo caso si poteva risparmiare spazio riunendo i due DRAW e trasformando il PSET in un BM

Come abbiamo già detto il disegno viene effettuato nel colore corrente di primo piano, ne segue che, per cambiare il colore di due disegni, basta intervallare le relative istruzioni DRAW con opportuni COLOR

Output del programma Spostamento

È anche possibile però cambiare il colore di primo piano direttamente con un comando del GML, e precisamente con il C <n>, dove n è il numero da 0 a 15 che identifica il colore scelto

Per avere una dimostrazione del funzionamento di C potete far girare di nuovo il programma SPOSTAMENTO (listato 3) dopo aver modificato la linea 110 in DRAW "BM120,91C1", il rombo verrà disegnato in nero anziché in bianco

La maggioranza dei comandi che abbiamo esaminato fino ad ora dipendono da un parametro, farebbe molto comodo poter definire questi parametri non direttamente, come abbiamo fatto in tutti gli esempi precedenti, ma attraverso una variabile alla quale assegnare un valore esternamente all'istruzione DRAW

La cosa in effetti è fattibile, ogni parametro presente in una stringa di comandi del GML può essere definito tramite una variabile mettendo al suo posto = <nome variabile>.

Attenzione al punto e virgola, non è un errore di stampa

Vediamo un esempio

```
10 SCREEN 2
20 PSET (128,96)
30 FOR COL = 1 TO 15
40 DRAW "C=COL
50 DRAW "NU20ND20NR20NL20
60 DRAW "NE15NF15NG15NH15"
70 NEXT
80 GOTO 80
```

queste linee disegnano una serie di stelle nei colori da 1 a 15, come si vede, alla linea 40 il colore del disegno viene variato tramite la variabile COL

Oltre che variabili numeriche, in una sequenza di comandi del GML possono venire inserite anche variabili di tipo stringa, che devono contenere a loro volta delle sequenze di comandi grafici. Il risultato è praticamente analogo alla chiamata di una subroutine

La variabile stringa va inserita nella sequenza facendola precedere da una X e seguire da un punto e virgola. Il programma che segue serve proprio a mostrare le modalità d'uso di questa nuova possibilità, il risultato che si ottiene è identico a quello dell'esempio precedente

```
10 SCREEN 2
20 PSET (128 96)
30 A$ = "NU20ND20NR20NL20"
40 B$ = "NE15NF15NG15NH15"
50 FOR COL = 1 TO 15
60 DRAW "C=COL XA$;XB$"
70 NEXT
80 GOTO 80
```

Per concludere questa carrellata sul GML non ci resta che vedere gli ultimi due comandi S e A. Il primo e in grado di modificare la grandezza delle figure disegnate, il secondo invece il loro orientamento

S cambia il cosiddetto fattore di scala (SF, Scale Factor), più questo è elevato più la figura, a parità di stringa generatrice, sarà grande

La modifica viene effettuata inserendo nella sequenza di comandi la lettera S seguita da un numero n compreso tra 1 e 255,

Listato 1

```
10 REM ***************
20 REM *             *
30 REM *   QUADRATO  *
40 REM *             *
50 REM ***************
60 REM
70 SCREEN 2
80 PSET (100.68)
90 DRAW 'R55D55L55U55'
100 GOTO 100
```

Listato 2

```
10 REM ***************
20 REM *             *
30 REM *    ROMBO    *
40 REM *             *
50 REM ***************
60 REM
70 SCREEN 2
80 PSET (70.96)
90 DRAW  E55F55G55H55"
100 GOTO 100
```

Listato 3

```
10 REM ***************
20 REM *             *
30 REM * SPOSTAMENTO *
40 REM *   CURSORE   *
50 REM *             *
60 REM ***************
70 REM
80 SCREEN 2
90 PSET (40,65)
100 DRAW 'R50D50L50U50'
110 DRAW  BM120.91"
120 DRAW 'E40F40G40H40'
130 GOTO 130
```

Listato 4

```
10 REM ***************
20 REM *             *
30 REM *   STELLE    *
40 REM *             *
50 REM ***************
60 REM
70 SCREEN 2
80 COLOR,15,15
90 CLS
100 PSET (128,96)
110 FOR I=1 TO 50
120 X=INT(RND(1)*50)+99
130 Y=INT(RND(1)*50)+71
140 C=INT(RND(1)*13)+1
150 PSET (X,Y)
160 DRAW  S=I:
170 DRAW  C=C:
180 GOSUB 210
190 NEXT
200 GOTO 200
210 DRAW NU9ND9NR9NL9
220 DRAW NE6NF6NG6NH6'
230 FOR K=1TO250
240 NEXT
250 CLS
260 RETURN
```

Listato 5

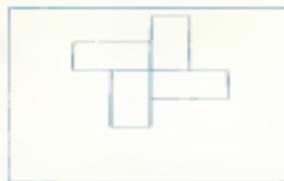
```
10 REM ***************
20 REM *             *
30 REM *  ROTAZIONE  *
40 REM *    ASSI     *
50 REM *             *
60 REM ***************
70 REM
80 SCREEN 2
90 PSET (120,70)
100 DRAW 'A0'
110 DRAW  R80D50L80U50'
120 GOSUB 230
125 CLS
130 DRAW  A1
140 DRAW  R80D50L80U50'
150 GOSUB 230
155 CLS
160 DRAW  A2
170 DRAW  R80D50L80U50'
180 GOSUB 230
185 CLS
190 DRAW  A3'
200 DRAW  R80D50L80U50
210 GOSUB 230
220 STOP
230 FOR I=1TO500
240 NEXT
250 RETURN
```

il fattore di scala diventerà uguale a n/4. Normalmente l'SF vale 1 (n=4) e si ha una corrispondenza 1 a 1 tra i numeri indicati nei comandi grafici ed i pixel dello schermo.

Il programma del listato 4 mostra una sequenza di stelle via via più grandi, ottenute sempre con gli stessi comandi, quello che cambia ad ogni iterata è appunto il fattore di scala (linea 160).

Il comando A, infine, ha la forma A <n>, con n compreso tra 0 e 4, ed il suo effetto è quello di ruotare lo schermo di 90•n gradi. A0 permette di ripristinare la condizione iniziale, ovvero rotazione nulla.

Il listato 5 mostra questo comando in funzione, il programma disegna quattro volte lo stesso rettangolo, ogni volta dopo una rotazione dello schermo.

Per quanto riguarda la grafica abbiamo concluso, l'appuntamento è alla prossima

Output del programma Rotazione usa: I rettangoli sembrano diversi sullo schermo a causa delle dimensioni dei pixel che non sono perfettamente quadrati.

puntata, nella quale ci occuperemo delle fatidiche sprite.

Ora è invece il momento di passare al software vero e proprio: questo mese vi presentiamo il programma Cargo, che inaugura la parte della rubrica dedicata al software dei lettori.

Cargo

di Pier Paolo Paoletti - Trieste

Cargo è un simpatico giochino, ispirato all'omonimo programma pubblicato esattamente un anno fa nella rubrica riservata al software per il TI 99/4A.

La trama è molto semplice: bisogna difendere le proprie astronavi dai colpi nemici, proteggendole con una sorta di scudo spaziale. Sembra facile, ma dopo i primi colpi la velocità di fuoco e di spostamento dell'astronave avversaria diventa tale da rendere Cargo un durissimo banco di prova per i propri riflessi.

A prescindere dal gioco in sè, il pro-

```
100 REM
110 REM          CARGD
120 REM
130 REM
140 REM    VERSIDNE  MSX
150 REM
160 REM Pier Paolo PADLETTI
170 REM
190 REM
200 REM —— TITDLI
210 CDLDR15,1,1:SCREEN2:KEYOFF
220 S=0:DRAW"C=S;':S=0
230 DRAW"0M40,20 D150 R100 U150 L100"
240 LINE(97,20)-(154,60),4,0F
250 DPEN"grp:"FDROUTPUTAS#1
260 PRESET(107,40),0:PRINT#1,"CARGO"
270 LINE(105,50)-(145,50),15
200 PRESET(92,65):PRINT#1,"Arcade by'
290 PRESET(57,05):PRINT#1,"Pier Paolo Pa
oletti"
300 FDRYY=46TD166STEP60
310 LINE(YY,100)-(YY+45,150),11,0F
320 CIRCLE(YY+23,125),21,0,,,1.3:PAINT(Y
Y+25,125),0
330 CIRCLE(YY+23,125),11,11,,,1.3:PAINT(
YY+25,125),11
340 NEXT YY
350 PRESET(44,102):PRINT#1,"Press any ke
y to begin"
360  IFINKEY$=""THEN360
370 GDSU0 1560
300 CLS:FORI=1TD2
390 PLAY'c...",'e... ,"g...'
400 PLAY f.',"a.",'c."
410 PLAY"b.","d.",'g.'
420 PLAY"f.","a.",'c.'
430 PLAY'c.","e. ,"g."
440 PLAY'c...", e.....", g.....'
450 PLAY"o514c :NEXT
400 REM —— DEFINIZIONE SPRITE
470 FDRS=0TD3:S$=""
400 FDRR=0TD7:READT$
490 S$=S$+CHR$(VAL("&b"+T$))
500 NEXTR:S$(S)=S$:NEXTS
510 SCREEN2,3
520 FDRN=0TD3:SPRITE$(N)=S$(N):NEXTN
530 REM ——  CARGD NEMICD
540 DATA 01111100
550 DATA 10101000
500 DATA 11111000
570 DATA 00111111
500 DATA 00111111
590 DATA 11111000
600 DATA 10101000
610 DATA 01111100
620 REM —— 0ARRIERA
030 DATA 11111111
040 DATA 00010000
650 DATA 01000100
660 DATA 00010000
670 DATA 01000100
000 DATA 00010000
690 DATA 01000100
700 DATA 11111111
710 REM —— CARGD AMICD
720 DATA 00000001
730 DATA 00111111
740 DATA 00000110
750 DATA 11111100
760 DATA 11111100
770 DATA 00000110
780 DATA 00111111
790 DATA 00000001
000 REM —— ——   LASER
010 DATA 00000000
020 DATA 00000000
030 DATA 00000000
040 DATA 11111111
050 DATA 00000000
060 DATA 00000000
070 DATA 00000000
000 DATA 00000000
090 CLS:SCREEN 0:LOCATE3,7:INPUT"Qual e'
 il tuo nome ";A$
900 REM —— INIZIO GIDCD......
910 SCREEN2:CDLDR15,4,1
920 PLAY"T255S11M5000o6L4Co50AGo6Co5L000
L4AGFL0FFL4GGAo6Co5L2M100000L4M5000EL0EE
;L4AAGAL2M100000-10m5000AGFE14DAGo6FC...
 ."
930 FDRI%=1TD150
940 X=RND(1)*255:Y=RND(1)*164:C=INT(RND(
1)*15)+1
```

continua a pagina 98

gramma presenta anche un elevato valore didattico per il buon uso che l'autore ha fatto delle caratteristiche grafiche e musicali dello standard MSX. Qua e la per il listato, che vi consigliamo di studiare con attenzione, si trovano spunti di notevole interesse: alle righe 250 e seguenti, ad esempio, viene mostrato come sia possibile scrivere del testo sullo schermo in alta risoluzione, argomento, questo, spesso ignorato dai manuali.

Vediamo come si svolge il gioco: caricato il programma e dato RUN compaiono i titoli di testa e le istruzioni, poi viene richiesto il nome del giocatore ed infine appare il campo di gioco, costituito da uno sfondo stellato, dalle tre astronavi da proteggere e dalle segnalazioni relative al punteggio.

Un attimo dopo compare la nave nemica, che apre subito il fuoco. La barriera difensiva va posizionata davanti all'astronave presa di mira agendo sui tasti del cursore. Se si riesce a sopravvivere fino ai 500 punti viene attribuito dal computer un bonus di valore casuale, nel caso bisogna stare attenti a non farsi distrarre dall'apparizione della scritta che segnala il fatto.

Prima di iniziare a giocare conviene inserire la condizione di CAPS LOCK, altrimenti il programma non riconoscerebbe la risposta ad alcune richieste di input. MC

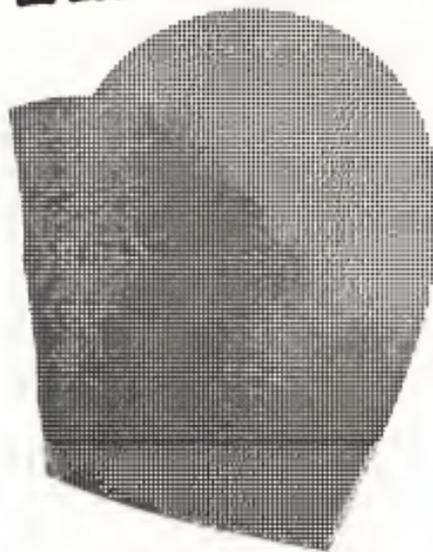
(segue da pagina 97)

```
950 PSET(X,Y),C:NEXT
960 SPRITEON
970 PRESET(15,165):PRINT#1,'H1 SCORE';HI
980 PRESET(05,105):PRINT#1,HI$
990 PRESET(160,165):PRINT#1,"SCORE"
1000 PRESET(160,175):PRINT#1,A$
1010 PRESET(15,175):PRINT#1,NO$
1020 FORJ=1TO500:NEXT
1030 PUT SPRITE 0,(220,40),9,2
1040 PUT SPRITE 1,(220,00),7,2
1050 PUT SPRITE 2,(220,120),3,2
1060 PUT SPRITE3,(200,00),0,1
1070 P=50:SC=0:XX=00:S1=0
1000 REM ———————— NUCLEO GIOCO
1090 X1=INT(RND(1)*3)+1
1100 IFX1=1THENX=40:GOTO1140
1110 IFX1=2THENX=00:GOTO1140
1120 IFX1=3THENX=120:GOTO1140
1130 GOTO1000
1140 0EEP:PUT SPRITE4,(15,X),4,0
1150 FORN=0TOP+10
1160 OUT170,(INP(170)AND240) OR0
1170 A=INP(169)XOR255
1180 IFA=16 THENXX=40
1190 IFA=32 THENXX=00
1200 IFA=120THENXX=120
1210 PUT SPRITE3,(200,XX),0,1
1220 NEXT N
1230 0EEP:0EEP:FORI=20TO200STEP10:PUT SP
RITES,(I,X),15,3:NEXT:PUT SPRITES,(200,X
),0,3
1240 IFX=XXTHEN GOSU01270
1250 IFX<>XXTHEN GOSU01350
1260 GOTO 1000
1270 REM ——— AGGIORNAMENTO SCORE
1200 SC=SC+10
1290 FORK=1TO9:IFSC=500*KTHENPRESET(90,1
85):PRINT#1,K;"8ONUS':FORH=1TO150:NEXTH:
PRESET(90,105):COLOR1:PRINT#1, [illegible]
:COLOR15:SC=SC+INT(RND(1)*10)*10
1300 NEXTK:SC$=STR$(SC)
1310 COLOR 1:PRESET(200,165):PRINT#1, [illegible]
[illegible]
1320 COLOR15:PRESET(208,165):PRINT#1,SC$

1330 IF P<=0THENRETURN
1340 P=P-1:RETURN
1350 REM ——— FINE MANCHE
1360 FORI=2TO15:0EEP
1370 COLOR I,1,15
1300 FORJ=1TO50:NEXT J
1390 COLOR15,15,1
1400 NEXT I:COLOR 15,1,1
1410 IFSC>HITHENHI=SC:NO$=A$
1420 FORJ=1TO500:NEXT
1430 SCREEN0:CLS:LOCATE15,4:PRINT"CARGO"

1440 LOCATE3,10:PRINT"HI SCORE..":HI
1450 LOCATE20,10:PRINT"SCORE..':STR$(SC)

1460 LOCATE3,13:PRINTNO$
1470 LOCATE20,13:PRINT A$,......
1400 PRINT"   GIOCHI ANCORA    (S/N) ?"
1490 K$=INKEY$:IFK$<>"N"ANOK$<>"S"THEN 1
490
1500 IFK$="N"THEN CLS:ENO
1510 IFK$="S"THEN PRINT:PRINT"   STESSO
GIOCATORE (S/N) ?":PRINT:PRINT
1520 K$=INKEY$:IFK$<>"N"ANOK$<>"S"THEN 1
520
1530 IFK$="N"THEN INPUT"   NOME";A$:CLS:
COLOR 15,1,1:GOTO900
1540 IFK$="S"THEN CLS:COLOR 15,1,1:GOTO9
00
1550 GOTO1520
1560 REM ——— NOTIZIE
1570 SCREEN0:LOCATE0,1
1500 PRINT"USA I TASTI CURSORE.":PRINT:P
RINT'          LEFT, RIGHT, NORTH."
1590 PRINT:PRINT"SEI IN VIAGGIO CON TRE
CARGHI E SEI A":PRINT"0ORO0 OELL' AMMIRA
GLIA, QUANOO VENITE":PRINT"ATTACCATI OA
UN CACCIA NEMICO.":PRINT
1000 PRINT"PUOI OIFENOERE LA FLOTTA OALL
'ATTACCO":PRINT"ACCENDENOO LO SCUOO ENER
GETICO OELLA ":PRINT"NAVE DI VOLTA IN VO
LTA SOTTO TIRO."
1610 LOCATE10,17
1620 PRINT'FAI ATTENZIONE !"
1630 LOCATE 2,19:PRINT"IL NEMICO E' SEMP
RE PIU' VELOCE !"
1640 LOCATE11,22:PRINT"Premi un tasto"
1050 IF INKEY$=""THEN1650
1060 SCREEN2:COLOR 4,1,1
1670 RETURN
```

Ama il meglio!

32K ROM 80K RAM
Tastiera professionale a 90 tasti
Porte per monitor, TV, joysticks, floppy disk, cassette recorder, stampante, giochi.
Interfaccia stampante parallela Centronics incorporata

il computer del grande standard MSX

Parla più FORTH

di Raffaello De Masi

Ottava parte

Un po' di pratica

Cammina, cammina, Ermanno Olmi ci ha fatto un film e noi siamo arrivati a specializzarci ed a manipolare stack, flag e word quali esperti nocchieri (ve lo ricordate il nauta della prima declinazione?) nel procelloso mare del Forth. Ma poiché le calde e quiete lagune del Basic e del Fortran non ci avevano mai dato il brivido dell'avventura ci siamo spinti in mare aperto e ormai gli amici di una volta chi li vede più?

Ma il viaggio è stato duro e lungo e stavolta è il caso di fermarci. Per cui, per questa puntata, niente nuove word, ma qualche utile applicazione di quanto abbiamo fatto finora, con qualche programmino di utilità sempre buono da tener registrato da qualche parte.

Già parlammo, un paio di puntate fa, in occasione dei cicli DO-LOOP, delle word LEAVE ed EXIT. Ricorderemo che la prima determina la chiusura di un ciclo DO-LOOP, identificando il valore del contatore al limite del loop, mentre la seconda agisce come una vera e propria word di chiusura (quindi come un [;]) pur avendo la limitazione di non poter essere usata con un ciclo DO-LOOP, cui, comunque, assolve la già citata LEAVE.

Evidentemente sia LEAVE che EXIT vanno precedute da una comparazione, altrimenti non avrebbero motivo di esistere. Comunque è uso comune che EXIT sia generalmente utilizzato per routine di salvataggio, come, ad esempio, in caso di errore, mentre LEAVE interviene quando un evento desiderato avviene.

La figura 1 riporta un programmino che consente di ricercare, in uno specifico blocco di memoria, un certo numero e, in ossequio a quanto precedentemente affermato, usa LEAVE per uscire dal loop quando viene trovato il numero stesso. Per consentire ciò, l'indirizzo ricercato viene portato in stack, ma, se il numero non viene trovato in TOS, viene riportato a zero.

All'inizio, si assume, evidentemente, che il numero non sia stato trovato per cui viene posto in TOS uno zero. Stabiliti i limiti di DO-LOOP, il programma legge un numero in memoria e lo compara con il valore di test. Se i due valori sono uguali (ricerca ultimata) un IF-THEN sostituisce lo zero dello stack con l'indirizzo effettivo del numero ed esegue un LEAVE forzando il ciclo DO-LOOP. Altrimenti si riprende daccapo.

Al termine del ciclo sia esso stato fruttuoso o no, i due numeri in cima allo stack, ind I e n, vengono cancellati lasciando solo l'indirizzo ricercato, in caso di ricerca positiva, o lo zero, in caso negativo.

Come già abbiamo detto EXIT è generalmente utilizzato come escape da situazioni d'errore come, per esempio, un input non valido.

La definizione, data la puntata precedente, della radice quadrata può essere, in caso di errore quale la richiesta di essa per un numero negativo, così integrata (dopo la word iniziale)

```
DUP O<
IF EXIT THEN
1
BEGIN
OVER OVER /

ecc
```

Ricerca ed ordinamento in array e tabelle

In molte applicazioni, i numeri di una array rappresentano i risultati di test, raccolta informazioni o, cosa per cui il linguaggio è particolarmente tagliato, sequenze di dati ottenuti da un controllo di una macchina o di una strumentazione. Se tali informazioni debbono essere successivamente analizzate, può essere necessario il loro riordino sia in senso numerico (crescente o decrescente) od alfabetico. La figura 2 e successive descrivono due comuni tecniche di riordino (o sorting), generalmente le più usate, chiamate bubble sort ed insertion sort, e forniscono listati adatti alla loro implementazione.

Bubble sort

La tecnica di bubble sort è così chiamata poichè determina una continua risalita dei numeri in memoria, come bolle d'aria nell'acqua.

Ad esempio, in una array monodimensionale, ogni numero viene paragonato, partendo dal primo numero della lista stessa, con il successivo.

Se il numero risulta essere più grande del successivo, si effettua uno scambio tra di essi. I due numeri successivi vengono comparati, scambiati se necessario e così via. Dopo un certo numero di tentativi il programma avrà riordinato in senso crescente tutti i numeri ed il numero più grande sarà "ribollito" al posto più alto.

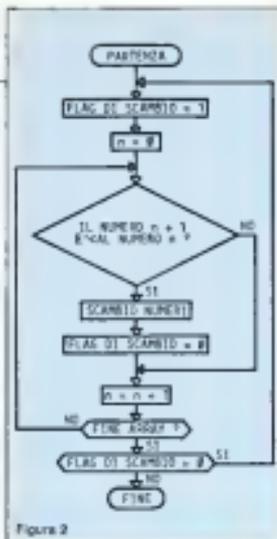
Questo algoritmo, universalmente conosciuto nel mondo dell'informatica ed applicato dappertutto in ogni linguaggio, richiede ovviamente diversi passaggi per completare il riordino. Purtroppo il numero di tali passaggi non è fisso, dipendendo esso dall'iniziale stato dell'array per cui il computer non avrebbe alcun termine di paragone per conoscere effettivamente quando interrompe l'operazione. È necessario, per tale ragione, utilizzare uno speciale indicatore che segnali se, in un passaggio, sia stato effettuato uno scambio. Tale indicatore, definito flag di scambio, avviserà il computer quando interrompere il sorting.

La figura 2 evidenzia la flowchart di bubble sort con uso del flag di scambio. Questo viene settato ad 1 all'inizio di ogni passaggio di sort. Viceversa ogni volta che un passaggio determina uno scambio di numeri il flag passa a 0. Se, pertanto, all'inizio di ogni ciclo di sort (o, che è la stessa cosa, alla fine della array) il flag presenta il valore 0, occorre ripetere l'operazione mentre per valore 1 (vero) di flag il sorting può essere ritenuto completo ed il programma esce dal loop.

Come si vedrà, l'operazione richiede almeno 1 sequenza di confronti (ipotesi minima per array già ordinata). Per una array messa nell'ordine esattamente inverso (caso più disastroso) occorrono (fatevi un pic-

colo conto od una prova con carta e matita) n-1 passaggi per il riordino totale +1 per la verifica finale. Vale a dire che, una array di N posti impiegherà da 1 ad N passaggi per il sorting con (N+1) /2 passaggi come media.

La figura 3 mostra la definizione della word BUBBLE, che consente appunto la bubble sort di una array di numeri (o di parole, tanto basta usare gli indicatori ASCII). Essa è facile da interpretare, seguendo precisamente la flowchart della figura 2. Notare lo uso di >R in coppia, che consente di salvare nel return Stack (lo ricordate, quanto tempo è passato!) i limiti di DO-LOOP, che rappresentano gli estremi della array in cui va effettuata la ricerca. Notare anche la sequenza [!!] che consente (il riordino dei parametri viene eseguito dal 4ROLL) di conservare il primo numero nel secondo indirizzo e viceversa, vale a dire consente lo scambio dei numeri se la condizione I 2 + @ I @ < (comparazione tra i due numeri) setta il flag a 1.

L'operazione eseguita dalla word BUBBLE ha però uno svantaggio. Ogni esecuzione del ciclo DO-LOOP causa il confronto di tutti i numeri in array, anche quelli che già sono stati scambiati e portati alla loro posizione finale. Poiché non c'è necessità di valutare ogni volta tali numeri, è possibile, ancora di più, accelerare l'operazione ammettendo, nell'ambito del programma, la comparazione di tali numeri. Vale a dire che, nel riordino dell'array, il primo passo analizzerà N numeri, il secondo N-1 numeri, il terzo N-2 e così via.

Pertanto, per definire una nuova word con un algoritmo più efficiente occorre modificare il programma precedente in modo da ridurre il limite di DO-LOOP di 2 ad ogni passaggio. La figura 4 mostra una versione migliorata della word BUBBLE, XBUBBLE in cui appunto il limite di DO-LOOP viene decrementato, ad ogni round, di 2, immediatamente dopo l'estrazione dal return stack. Inoltre l'indirizzo iniziale, per consentire effettivi n passaggi è incrementato di 2. Tutto il resto è uguale.

È davvero più veloce tale nuova definizione? Analogamente a quanto suggerisce Scanlon, abbiamo anche noi eseguito una prova su una array di 300 numeri messi esattamente al contrario (in senso decre-

PARTENZA
FLAG DI SCAMBIO = 1
n = Ø
IL NUMERO n + 1 E'<AL NUMERO n ?
NO
SI
SCAMBIO NUMERI
FLAG DI SCAMBIO = Ø
n = n + 1
NO
FINE ARRAY ?
SI
FLAG DI SCAMBIO = Ø
SI
NO
FINE

Figura 2

```
: RICERCA
  ( consente la ricerca in memoria di un numero )
  ( tra gli indirizzi ind1 ed ind2 )
  ( se n e' trovato , il suo indirizzo e' in TOS )
  ( altrimenti sara' presente 0 )
  ( ricerca positiva : ind1 ind2 n - - -  ind )
  ( ricerca negativa : ind1 ind2 n - - - 0 )
  0             ( all'inizio , poniamo in TOS 0 )
                ( vale a dire , immaginiamo di )
                ( non aver trovato il valore cercato )
  4 ROLL ROT    ( riordinando cosi' in Stack )
  4 ROLL        ( 0 , ind 1 , n , ind 2 )
  3 PICK -
  2+
  0             ( impone l'inizio di DO-LOOP )
  DO
   OVER I+
   @
   OVER =
   IF
   ROT 3 PICK + I +
   SWAP ROT
  THEN
 2
 +LOOP          ( per la sequenza , vedi testo )
 DROP DROP      ( cancella i due numeri in TOS )
                ( fine definizione )
```

Figura 1

```
: BUBBLE
  ( consente il riordino di una array di n numeri )
  ( con inizio all'indirizzo in TOS )
  ( indirizzo n - - - )
  DUP + OVER + 2 -     ( limite del DO-LOOP )
  SWAP                 ( inizio del DO-LOOP )
  >R >R                ( conserva i limiti nel Return Stack )
                       ( vedi testo )
  BEGIN
  1
  R > R@ OVER >R       ( cerca i limiti di DO-LOOP )
  DO
   I 2 + @ I @ <       ( compara i due numeri successivi )
   IF                  ( se il 2^ numero < 1^ )
                       ( scambia i due numeri )
   I 2+ DUP @
   I DUP @
   4 ROLL ! !
   DROP 0              ( metta il flag di cambio a 0 )
  THEN
  2
 +LOOP
 UNTIL                 ( ricomincia finche' flag = 1 )
 R> R> DROP DROP       ( elimina i limiti di DO-LOOP )
                       ( fine definizione )
```

Figura 3

scente) I risultati sono stati i seguenti (in sec)

	BUBBLE	XBUBBLE
Computer APPLE IIE		
Forth su scheda	147	101
Forth su disco	159	105
Computer HEWLETT-PACKARD 87		
Forth su disco	96	68
Computer ITT 3710		
Forth su disco	125	92
Computer IBM PC		
Forth su scheda	98	71

con un vantaggio medio, quindi, del 25,30%.

È interessante anche notare come, prevedibilmente, il linguaggio su scheda sia più rapido di quello su disco e come l'HP nonostante il suo processore ad 8 bit (si tratta comunque di un sistema multiprocessore) sia altrettanto rapido dell'IBM con il suo 16 bit, dimostrando l'alta vocazione scientifica della serie 80

Esiste però un'altra tecnica di sorting, ben più rapida, pur essendo altrettanto semplice Si tratta dell'Insertion Sort, anche questa tecnica è abbastanza semplice, anzi più intuitiva ancora delle precedenti In effetti è quella che usiamo comunemente quando riordiniamo un mazzo di carte Infatti, partiamo dalla 2ª carta e la confrontiamo con la prima, eseguendo il normale bubbling, poi, passati alla successiva, ognuna di esse viene comparata con tutte le precedenti e risale nel gruppo fino al suo effettivo posto relativo prima di procedere con un altro numero Si esegue, pertanto, proprio l'inserzione che si compie quando si riordinano le carte in mano durante una partita di canasta o di un più casereccio scopone La figura 5 mostra la definizione di ISORT, si notino i due cicli DO-LDOP nidificati, quello più esterno scorre attraverso l'array partendo col secondo numero, e comparando ogni membro dell'array con quello precedente Se occorre lo scambio [N(n-1) > N (n)] si passa al DO-LDOP interno che ricerca il definitivo posto del numero nell'array, lasciando inoltre un indirizzo nello stack Inoltre il loop interno (DUP DUP 2+) apre uno spazio nell'array e, successivamente (1 3 PICK — <CMOVE !) inserisce il numero, come si fa appunto quando si riordinano le carte in mano

Le stesse prove effettuate precedentemente sono state eseguite in tempi, con questa tecnica, ancora ridotti del 35%, circa con rapporti più favorevoli per l'HP e l'IBM e con un collasso dell'Apple che, specie operando su disco, ha mostrato ridotte differenze con la tecnica precedente

Amici Forthisti, che spero (non per mio merito, s'intende) ben più numerosi del centinaio di iscritti al FIG Italia, avete visto come è necessario avere a disposizione qualcosa di più efficiente e rapido del solito Basic o del sussiegoso Pascal Proviamo a fare quello che abbiamo descritto in un altro linguaggio, e vedremo chi taglia il traguardo Certo, non è facilissimo lavorarci, ma avete mai visto una corsa di formula 1 o meglio ancora una 24 ore di Le Mans? Il vincitore, alla fine della corsa, non è mica da prima pagina di Capital e non profuma certo d'acqua di Vichy, ma è arrivato prima e fa la doccia agli spettatori col magnum di champagne Chi va in B od in P sta sotto, tra la folla, applaude, tocca con un dito le Ferrari come fossero reliquie di papa Giovanni, se tutto gli va bene rimedia un autografo di Arnoux o di Lauda, e torna a casa a vedere la domenica sportiva **MC**

```
: XSORT
  ( versione piu' efficiente di SORT )
  ( vedi testo )
  ( indirizzo n  - - - )
  DUP + OVER +    ( fine DO-LOOP = ind + 2n )
  SWAP            ( partenza DO-LOOP = indirizzo )
  >R >R           ( salva tali valori nel Return Stack )
  BEGIN
   1
   R> 2 - R@ OVER >R     ( sottrae 2 dal limite finale )
   DO
    I 2 + @ I @ <        ( compara i due numeri seguenti )
    IF
     I 2+ DUP @
                         ( se il 2^ numero < del 1^ li scambia )
     I DUP @
    4 ROLL               ( vedi testo )
    DROP 0               ( setta il flag di cambio )
   THEN
   2
   +LOOP
  UNTIL                  ( continua finche' il flag = 1 )
  R> R> DROP DROP        ( cancella i limiti di DO-LOOP )
                         ( fine definizione )
```

Figura 4

```
ISORT
( usa la tecnica dell'insertion sort )
( indirizzo n  - - - )
DUP + OVER +          ( fine del DO-LOOP )
OVER 2+               ( inizio del DO-LOOP)
 DO
 I @ DUP         ( controlla se l'n numero e' < del precedente)
 I 2 - @ <
  IF
   0
   3 PICK 2 - I 2 - (  limiti del loop interno = )
                    (  indirizzo - 2 ; I - 2 )
   DO
    DROP
    DUP I @ <
    IF
     I            ( se le condizioni si verificano,salva l'ind. )
    ELSE
     I 2+ LEAVE
    THEN
    -2
   +LOOP
   DUP DUP 2+          ( apre uno spazio per l'inserzione )
   1 3 PICK - <CMOVE    ( ed inserisce il numero)
  ELSE
   DROP                ( cancella il numero )
  THEN
  2
 +LOOP
DROP                   ( pulisce lo Stack )
                       ( fine definizione )
```

Figura 5

di Pierluigi Panunzi

Le istruzioni di salto

Dopo aver vista nelle scorse puntate un gran numero di istruzioni "statiche" nel senso che non portavano altro "sconvolgimento" che la lettura di un certo dato dalla memoria, ci occuperemo questa volta di un gruppo di istruzioni molto importanti in quanto consentono di alterare la naturale sequenza di un programma stiamo parlando delle cosiddette "istruzioni di salto".

In presenza di tali istruzioni lo Z80 invece di eseguire la istruzione successiva a quella eseguita per ultima, salta ad un'altra punta della memoria dove troverà ad aspettarlo (se tutto va bene!) un altro pezzo di programma.

Analizzeremo dunque vari tipi di istruzioni di salto dapprima ci occuperemo dei salti assoluti (condizionati e non), poi dei salti cosiddetti relativi (anche questi condizionati e non) ed infine di alcune particolari istruzioni di salto indiretto e di gestione dei cicli di istruzioni.

Come dice il loro nome e come e facilmente intuibile, le istruzioni di salto consentono al programma di saltare da un punto all'altro della memoria oppure, ma e la stessa cosa, permettono al programmatore di far eseguire determinate parti di un certo programma invece di altre, per il fatto che si e verificata una certa condizione, o per il semplice motivo che è necessario effettuare il salto

Le istruzioni di salto per lo Z80 si dividono in tre gruppi al primo appartengono i salti, condizionati e non, assoluti, al secondo i salti relativi, anche questi condizionati e non, mentre al terzo appartengono quelli indiretti Cominciamo allora dal primo gruppo per dire che il loro attributo di "assoluti" deriva dal fatto che l'istruzione stessa specifica l'indirizzo di memoria al quale deve saltare il programma in particolare si tratta di istruzioni formate da tre byte, in cui il primo e il codice operativo ("Opcode") e gli altri due rappresentano l'indirizzo di salto

Soffermiamoci su questo indirizzo con un esempio iniziamo con l'istruzione "JP 1234H", che impone al programma di saltare all'indirizzo (esadecimale) 1234H

Supponendo che l'istruzione si trovi all'indirizzo (sempre esadecimale, ma ora non lo diremo più) 0500H automaticamente lo Z80 si posizionerà sull'indirizzo 1234H, praticamente ignorando 0C31H celle di memoria

Non perdiamo l'abitudine a fare piccoli calcoli in esadecimale! Il valore citato si ottiene semplicemente effettuando la sottrazione tra 1234H e 0500H, che da 0C34H ora però da questo valore dobbiamo sottrarre i due byte rimanenti dell'istruzione di salto che stiamo analizzando ed infine dobbiamo ancora sottrarre 1 in quanto stiamo calcolando il "numero di byte" tra l'istruzione di partenza e quella di arrivo

Chiudiamo questa parentesi, in quanto generalmente non dobbiamo fare questi calcoli ci serviva soltanto per far vedere l'"entità" di un certo salto (nel nostro caso "assoluto"), entità che può assumere praticamente qualsiasi valore tra 0000H ed FFFFH, sottintendendosi così che con un'istruzione di salto assoluto si può raggiungere qualsiasi punto della memoria, senza alcun tipo di "impedimento" o "protezione" che dir si voglia

Detto ciò possiamo analizzare le modalità di esecuzione dell'istruzione in particolare l'istruzione vista fa saltare sempre all'indirizzo dato

Esistono poi le istruzioni di "salto assoluto condizionato", per le quali giocano un ruolo molto importante i "flag", dei quali abbiamo parlato abbondantemente nella puntata relativa alle operazioni logico-aritmetiche dello Z80

Non a caso ci eravamo soffermati sui tre flag C, Z e S (rispettivamente il Carry, il flag di Zero e quello di Segno) questa volta non ci occuperemo del quarto flag (N), ma viceversa parleremo dell'altro flag (P, flag di Parità)

In particolare, nel caso di un'istruzione di salto condizionato bisogna far bene attenzione al valore di questi quattro flag ed in particolar modo a quello strettamente legato all'istruzione di salto

Dato che i flag sono quattro (ripetiamo Z, C, P, S) e dato che ognuno di questo può avere due "stati" (ne riparliamo più oltre ...) ecco che otteniamo 8 istruzioni di salto, che sono

```
JP  Z,nnnn
JP NZ nnnn
JP  C,nnnn
JP NC,nnnn
JP PO,nnnn
JP PO nnnn
JP PE nnnn
JP  P,nnnn
JP  M nnnn
```

dove "nnnn" e un indirizzo assoluto esadecimale, formato da 4 cifre esadecimali (come ad esempio 1000H, D4F3H, 777AH ecc)

Analizziamo la prima istruzione, che si legge "salta, se zero, all'indirizzo nnnn" come dice la frase, il salto avviene se e solo se dalle istruzioni precedenti, ed in particolare "la" precedente, si ha che il flag "Zero" e settato

Se ciò non accade, e questo vale in *tutti* i casi di salto condizionato, e cioè se la condizione non e verificata, allora l'istruzione di salto viene semplicemente ignorata ed il programma passa all'istruzione successiva

Nella seconda istruzione dell'elenco il salto avviene se e solo se il flag di Zero e 0 (ad esempio se il risultato di una certa operazione "non" era zero, secondo l'ormai consueto gioco di parole)

Analogamente si ha per la terza e la quarta istruzione, che si riferiscono stavolta al flag di Carry

La quinta e la sesta istruzione si riferiscono invece al flag di Parità vediamo ora più in dettaglio il significato di questo flag

Dato un certo byte, che come sappiamo bene è formato da 8 bit, si intende per parità la somma dei bit "1" costituenti il byte stesso il flag o bit di parità è concepito in modo tale da rendere sempre "dispari" il numero totale di "1" (8 del byte, più uno dato dal flag stesso)

Facciamo un esempio, anzi un paio!

Supponiamo di avere il byte 34H esprimendo tale valore in notazione binaria avremo 00111000 .

Ora andiamo a contare il numero di uni del nostro byte: è facile! Sono tre. Ora il flag di parità sarà in questo caso 0 in quanto solo così il totale degli uni rimane dispari. Viceversa, consideriamo il byte F0H scritto in binario abbiamo 11110000. In totale abbiamo dunque quattro uni: avendo a disposizione l'ulteriore bit dato dal flag di Parità, questo deve valere "1" rendendo così dispari (pari a 5) la somma dei bit pari ad "1".

Nel primo degli esempi (in cui P vale 0) si dice che la parità è "dispari" ("Odd" in inglese), invece nel secondo esempio la parità si dice "pari" ("Even" in inglese) ed il flag P è pari ad 1.

Tornando perciò alle istruzioni di salto condizionato, le "JP PO,nnnn" e "JP PE,nnnn" si leggono "salta se la parità è dispari..." e "salta se la parità è pari...".

Evidentemente sarebbe meglio leggere le istruzioni in inglese "jump on parity odd to nnnn" e "jump on parity even to nnnn" in quanto solo così abbiamo la corrispondenza con le sigle PO e PE. A loro volta, comunque si leggano, le istruzioni significano: la prima, "salta se P = 0" e la seconda "salta se P = 1".

A questo punto però si impone una fermata: specie per chi è alle prime armi, non è molto facile distinguere subito bene concetti come parità pari, dispari e collegarli a PO, PE e ai valori di P: proponiamo dunque un semplicissimo metodo mnemonico (inventato dall'autore dell'articolo, ma facilmente scovabile!), che per il fatto di essere mnemonico è, sì, molto semplice, ma altresì non trova alcun riscontro "matematico". Eccolo dunque: per raccapezzarcisi sulle JP PO e JP PE basta semplicemente fissare l'attenzione sulla prima (per l'altra si va evidentemente ad esclusione!), che è "JP PO".

Le lettere "PO" come visto significano "Parity Odd" per un caso fortuito la chiave è nella paroletta "Odd" dove c'è praticamente tutto! C'è la "O" che ci ricorda che il bit di parità è "0" ed in più ci sono ben due "d" che ci indicano che la parità è "d-ispari"! Semplice ma a suo modo efficace!

Ripetiamo che dobbiamo solo pensare alla sigla PO e alla parola "Odd" inevitabilmente il discorso opposto non si può ottenere da PE e tantomeno da "Even" (forse, machiavellicamente, si potrebbero contare le cifre che compongono le "parolette": è un caso anche questo, ma "Odd" è formato da tre lettere, mentre "Even" da quattro!!). E poi dicono che la matematica non è un'opinione...

Comunque lasciamo al lettore la scelta di quale metodo usare per associare meccanicamente o coscientemente i vari concetti relativi alla parità: altra scelta è anche quella di ignorare completamente il nostro "truecchetto ignobile"...

Tornando dunque alle ultime due istruzioni di salto condizionato, abbiamo le "JP P,nnnn" e "JP M,nnnn" rispettivamente si riferiscono al segno risultante dalle operazioni precedenti. Se il bit di flag S è 1, il segno negativo (il segno, in inglese è "Minus") e viceversa, se S è 0, il segno è positivo (in inglese "Plus"). Ecco che la prima istruzione effettuerà il salto se il segno risultante era "Plus", mentre la seconda lo effettuerà se il segno risultante è "Minus".

I salti relativi

È questo un tipo di istruzione che si trova anche in altri microprocessori ad 8 bit, quale ad esempio il 6502, ma che viceversa non era presente nell'8080, il processore dello Z80. Come dice il nome, si tratta di salti "relativi all'indirizzo" di memoria in cui si trovano: con queste istruzioni si può saltare ad una certa istruzione che si trova, rispetto a quella di salto, nelle "immediate vicinanze" intendendo con questo termine una distanza di al massimo 127 byte in più oppure 128 byte in meno, rispetto alla locazione in cui si trova nell'istruzione di salto.

Come si vede, dunque, c'è una differenza sostanziale con le istruzioni di salto precedenti: in questo caso infatti l'indirizzo di arrivo *non* può assolutamente spaziare tra 0000H ed FFFFH, come avveniva invece nel caso precedente.

La spiegazione di ciò ed in particolare del valore massimo del "range" di indirizzi raggiungibili, risiede nel fatto che questo tipo di istruzioni è a due byte in cui il primo è al solito l'"opcode", mentre il secondo byte rappresenta un "valore esadecimale complemento a 2". Consigliamo a questo punto i lettori di fare riferimento alla puntata apparsa sul n. 36 di MCmicrocomputer: questa volta diciamo solo che secondo tale "logica", un byte può rappresentare un valore per l'appunto compreso tra -128 e +127.

In questo caso però, rispetto alle istruzioni di salto assoluto, abbiamo meno possibilità, fondamentalmente per una scelta dei costruttori; le istruzioni di salto relativo sono:

```
JR   e
JR   C,e
JR   NC,e
JR   Z,e
JR   NZ,e
```

dove "e" (in inglese "extension", estensione) rappresenta appunto il secondo byte.

La prima istruzione della tabellina rappresenta un salto relativo incondizionato, mentre le altre sono rispettivamente condizionate allo stato del flag di Carry (se è settato o no) e del flag di Zero (se è settato o no).

Non ci dilunghiamo oltre in quanto, a parte le considerazioni sull'indirizzo, non vi è altro di nuovo rispetto alle precedenti istruzioni di salto.

Le istruzioni di Salto Indiretto

Appartengono a questo piccolo gruppo tre istruzioni di salto indiretto

```
JP (HL)
JP (IX)
JP (IY)
```

nelle quali l'indirizzo non è indicato dall'istruzione stessa, ma viceversa è contenuto nella coppia di registri HL oppure nei registri indice IX e IY, dato che in ogni caso si tratta di registri a 16 bit (perciò 2 byte), abbiamo in questo caso le stesse possibilità di indirizzamento delle istruzioni assolute.

Vantaggio notevole di queste istruzioni è che l'indirizzo può essere "calcolato" dal programma stesso: in genere vengono sfruttate quando da una parte del programma possiamo saltare ad un certo numero di parti della memoria, parti i cui indirizzi iniziali sono ad esempio posti in un'apposita tabella.

Un'istruzione per i loop

Chiudiamo questa puntata riguardante i salti con un'istruzione molto potente, la

```
DJNZ e
```

la quale consente un'agevolissima gestione dei cicli o loop che dir si voglia.

In termini tecnici, si tratta di un'istruzione "multipla", nel senso che non si limita semplicemente ad effettuare un salto, seppur condizionato o no.

In questo caso l'istruzione decrementa il contenuto del registro B ed effettua un salto relativo (nell'istruzione compare infatti una "e") se il contenuto di B risultante non è zero; ecco dunque l'utilizzazione della DJNZ nei loop: basta porre nel registro B il numero di volte che il ciclo deve essere effettuato.

Facciamo dunque riferimento ad una situazione del genere

```
programma

LD B,numero di volte

(etichetta)
blocco di istruzioni
DJNZ etichetta
```

In questo caso abbiamo un programma in cui ad un certo punto poniamo in B un valore (compreso tra 0 e 255) ora si avrà che il "blocco di istruzioni" verrà eseguito tante volte quanto è il valore posto inizialmente nel registro B.

Infatti l'istruzione DJNZ decrementerà il registro B e fino al suo successivo annullamento effettuerà il salto all'"etichetta".

Dimenticavamo di dire che il nome dell'istruzione DJNZ deriva da "Decrement and Jump on Not Zero" e cioè "decrementa (il registro B) e salta sulla condizione di non-zero".

Chiudiamo dunque questa rassegna sulle istruzioni di salto dicendo che in nessuno dei casi visti si ha un'alterazione di alcuni flag, come dire che lo stato dei flag viene integralmente mantenuto, anche in quest'ultimo caso in cui il flag di Zero viene effettivamente utilizzato: ripetiamo, perché è importante, che lo stato dei flag rimane quello che si aveva "prima" dell'esecuzione dell'istruzione di salto, qualsiasi essa sia.

Nella prossima puntata ci occuperemo della gestione delle subroutine e vedremo che ritorneranno "a galla" molti dei concetti espressi questa volta. MC

SHIFT RUN PLAY GOAL!

__Calcio Computer__, 8 programmi software per Commodore 64, che di volta in volta ti insegneranno a compilare la schedina (e possibilmente vincere), elaborare i tuoi pronostici sulla classifica di campionato, raccogliere e consultare i dati delle squadre e dei giocatori di serie A e B.
Con __Calcio Computer__ si inaugura __Mondosoft__, il nuovo periodico per home computer della Mondadori.
Ogni quattordici giorni in edicola con un nuovo programma su cassetta corredato da un fascicolo a colori.
__Mondosoft__, per esplorare i vari "mondi" del software.

MONDOSOFT

Il periodico che leggi con il computer.

VIC da zero +64

di Tommaso Pantuso

Salviamo lo schermo

Nell'articolo di questo mese metteremo insieme tutte le nozioni apprese nei due numeri precedenti, riguardanti la memoria video e il buffer della tastiera, e cercheremo di trovare un metodo per codificare in maniera intelligibile a tutti, ciò che appare sul video, ad esempio i disegni che utilizzano i caratteri grafici Commodore che non sempre è facile interpretare quando, integrati in un programma, servono a produrre una schermata complessa. Inoltre avremo modo in seguito di applicare quanto sappiamo sul video per ottenere la memorizzazione su disco di intere schermate.

Un breve riepilogo

Abbiamo visto che esiste in memoria Ram, sia del Vic che del 64, una zona che riproduce, con un opportuno codice, tutto quanto compare sullo schermo. In pratica, se ad esempio sul video è possibile posizionare 1000 caratteri (40 colonne × 25 righe), in memoria troveremo una zona di 1000 byte, ciascuno dei quali corrisponde ad una determinata locazione dello schermo, contenenti dei numeri che è abbastanza facile mettere in corrispondenza con quanto effettivamente vediamo rappresentato. Ad esempio alla lettera A corrisponde il numero 1, alla B il numero 2 e così via. Il processo è naturalmente invertibile, nel senso che se noi andiamo a scrivere nella memoria video (ad esempio con una Poke) un numero che rappresenta il codice di un carattere, il carattere rappresentato comparirà sullo schermo in una certa posizione facilmente individuabile. In effetti abbiamo osservato che, per vedere effettivamente qualcosa, dobbiamo assegnare al carattere in questione un colore, cosa semplicemente ottenibile memorizzando un opportuno codice in un'altra zona Ram, la memoria di colore. In definitiva è come avere un sandwich a tre strati: il primo rappresenta lo schermo TV, il secondo la memoria video e il terzo quella di colore; per ottenere un carattere sul primo strato bisogna definire gli opportuni parametri sugli altri due.

Abbiamo anche visto l'importanza che assume quella zona di memoria chiamata Buffer di tastiera che, usata opportunamente, permette alla macchina, in un certo senso, di autoprogrammarsi in quanto è possibile, durante lo svolgimento del programma, aggiungere delle nuove linee senza arrestare l'elaborazione.

Vediamo allora ciò che possiamo produrre con tutte le nozioni accumulate negli ultimi articoli.

Schermo e "DATA"

Se possiamo scrivere in memoria video ed ottenere un'immagine sullo schermo, possiamo allora evitare di utilizzare l'istruzione Print. Ciò torna utile, come accennavamo prima, quando si vogliono comporre dei disegni la cui codifica, in un listato, sia facilmente interpretabile e quindi riproducibile. Cosa c'è allora meglio dei numeri per ottenere quanto vogliamo? Spieghiamoci meglio prendendo come punto di riferimento — senza perdere di generalità — il C 64, ritenendo abbastanza immediate le modifiche da effettuare sul Vic che in definitiva sono legate solo al formato dello

```
0 REM --- QUESTO PROGRAMMA INTRODUCE AUTOMATICAMENTE -
1 PEM ---   OELLE LINEE D1 DATA CHE CODIFICANO IL    -
2 REM --- CONTENUTO O1 UNA SCHERMATA DEL C 64        -
3 POKE52,156 POKE56,156 POKE650,128
5 PRINT"J".
6 POKE53280,0 POKE53281,0 PRINT"■"
10 GETA$ IFA$=""THENGOSUB100 GOTO10
30 IFA$="■"THEN GOSUB 200              REM TASTO F1
40 IFA$=CHR$(34)THEN10                 REM VIRGOLETTE
50 IFA$="■"THENGOSUB500                REM TASTO F3
60 IFA$="■"THENPRINT"J"·GOTO 1000 REM TASTO F5
90 PRINTA$  GOTO10
98 ENO
99 REM ------------------------------------------------
100 PRINT"● ■'.
110 FOR1=1TO50  NEXT
120 PRINT"■ "
130 PEM FOR1=1TO100 NEXT
140 PRINT"■'
150 RETURN
151 REM -----------------------------------------------
200 FORI=0TO959
210 POKE39936+I,PEEK(1024+I)
230 NEXT
240 PETURN
241 REM -----------------------------------------------
500 FOR1=55296TO56295
510 POKE1 1 NEXT
520 FORI=0TO959
530 POKE1024+I, PEEK(39936+I)
540 NEXT
550 RETURN
551 REM -----------------------------------------------
1000 REM -------- AUTODATA ---------
1010 REM ------------------------------
1020 POKE252,0
1030 REM PRINT"J".
1040 A=39936+16*PEEK(252)
1050 PRINTA "D♠ "
1060 FORS=0TO15
1070 A$=A$+PIGHT$(STP$(PEEK(A+S)),3)+" "
1080 NEXT PRINTA$+"■ ".
1090 POKE252 PEEK(252)+1
1100 POKE631,145 POKE632,145 POKE633,13
1110 POKE634,71 POKE635,207·POKE636,48
1120 POKE637,49 POKE638,52 POKE639,48
1130 POKE640,13 POKE198,10 END
1140 IF39936+16*PEEK(252)<39936+16*60THEN1030
1150 GOTO10
```

schermo ed alla diversa mappatura dello schermo in memoria

Supponiamo di voler scrivere nella prima locazione dello schermo quella più in alto a sinistra, il simbolo delle picche (quello delle carte). L'istruzione da dare in un eventuale programma sarebbe

PRINT" <Shift + A> "

Ciò naturalmente vale anche se si vuol scrivere qualunque altro carattere o stringa di caratteri grafici ma, mentre alcuni dei caratteri che compaiono nei listati sono facilmente individuabili sulla tastiera, altri lo sono un po' meno e quando essi compaiono in numero massiccio a volte non si riesce ad ottenere esattamente l'effetto desiderato dopo aver ricopiato un listato su cui sono presenti delle schermate grafiche introduttive o altre cose del genere

Dato che ormai siamo pratici di memoria di schermo, viene abbastanza logico pensare che lo stesso effetto potrebbe essere ottenuto andando a scrivere nella opportuna locazione il codice delle picche (65) ed abilitando convenientemente il colore. In altre parole, con

POKE 1024 65

(supponendo per comodità già abilitato il colore) vedremo comparire in alto a sinistra il carattere desiderato senza aver fatto nessun riferimento esplicito al simbolo grafico. Vi ricordiamo che da 1024 inizia la memoria di schermo per il 64

Se poi la stringa è più complessa, le cose non cambiano affatto in quanto basterà scrivere in tutte le locazioni interessate il codice del carattere desiderato. Ad esempio.

POKE1024 83 POKE1025 90
POKE1026 88 POKE1027 65

farà comparire, nelle prime quattro locazioni dello schermo, rispettivamente i caratteri cuori, quadri, fiori e picche, (supponendo sempre di aver abilitato il colore)

Naturalmente risulta abbastanza scomodo andare a ricavare dalle tabelle i codici di ciascun carattere che compone il disegno che vogliamo ottenere e scriverlo con delle Poke nelle precise locazioni di memoria, ma non è esattamente questo che noi vi proponiamo. Se potessimo ottenere, partendo da un immagine dello schermo in memoria, dei codici introdotti automaticamente in un certo numero di linee Data, potremmo poi facilmente da questi ricostruire lo schermo andando a "pokare" i dati nelle locazioni di schermo interessate. Ci spieghiamo meglio

Considerando l'esempio precedente, per quanto abbiamo detto, partendo dai simboli cuori, quadri, fiori e picche scritti nelle prime quattro locazioni dello schermo, dovremmo poter ottenere automaticamente nel programma, partendo dalla stringa in questione, una linea

DATA 83 90 88 65

attraverso la quale potremmo ricostruire le prime quattro locazioni di schermo con un segmento del tipo

```
10 FOR I = 0TO3 READ A POKE1024 + I,A
20 DATA 83 90 88 65
```

Naturalmente il processo è riconducibile a tutto lo schermo nel senso che se codifichiamo mille numeri con delle linee Data, cambiando il ciclo For... Next, avremo accesso a tutte le locazioni immagine

Questo processo, per essere comodo dovrebbe essere automatizzato a partire da una schermata nel senso che, una volta composto il disegno sullo schermo, da questo dovrebbero essere ricavate delle linee Data e introdotte automaticamente nel programma. È proprio l'implementazione di questo procedimento che vogliamo spiegarvi nelle righe che seguono.

AutoDATA

Il processo sintetizzato lo chiameremo d'ora in poi AutoDATA. Esso è ottenibile con una certa disinvoltura impiegando tutte le nozioni apprese sul comportamento del Buffer di tastiera in certe situazioni. Per meglio comprendere il procedimento d'insieme, facciamo riferimento alle linee che vanno da 1000 a 1150 del solito programma dimostrativo riportato nel listato 1 con il quale è possibile comporre un disegno o delle scritte sullo schermo e poi trasformare il contenuto della sua immagine in memoria in linee di dati. Ma cominciamo a commentare l'ultima parte, racchiusa tra le linee indicate e che rappresenta il cuore del programma, considerandolo un programma a sé

Vogliamo inserire in un certo numero di

```
39936 DATA  32,120,120,120,120,120,120,120,120,120,120,120,120,120, 97, 32
39952 DATA  32, 30, 32, 32,110,120,120,120,120,120,120,120,120,120,120,120
39968 DATA 120,120,120,120,120,120, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
39984 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 30, 32, 32,110, 32, 32, 32
40000 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40016 DATA  32, 32, 32, 32,  3, 21,  3,  9, 14,  1, 32, 32, 32, 32, 32, 32
40032 DATA  32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40048 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
40064 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,110, 32, 32, 32
40080 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40096 DATA 110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40112 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40128 DATA  32, 32, 32, 32, 32, 32, 32, 32,110,121,121,121,121,121,121,121
40144 DATA 121,121,121,121,121,121, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
40160 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40176 DATA 110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40192 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40208 DATA  32, 32, 32, 32, 32, 32, 97, 32,110, 32, 32, 32, 32, 32, 32, 32
40224 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
40240 DATA  32, 32, 19,  1, 12, 15, 14,  5, 32, 32, 32, 32, 32, 32, 32, 32
40256 DATA 110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40272 DATA  32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40288 DATA  32, 32, 32, 32, 32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32
40304 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,110, 32, 32, 32
40320 DATA  32, 32, 32, 32, 30, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40336 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40352 DATA  32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40368 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
40384 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32,110, 32, 32, 32
40400 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40416 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40432 DATA  32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40448 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
40464 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32,110, 32, 32, 32
40480 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40496 DATA  32, 32, 32, 32, 32, 12,  5, 20, 20, 15, 32, 32, 32, 32, 97, 32
40512 DATA  32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40528 DATA  32, 32, 32, 32, 32, 32, 97, 32,110, 32, 32, 32, 32, 32, 32, 32
40544 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32,110,120,120,120
40560 DATA 120,120,120,120,120,120,120,120,120,120,120,120,120,120, 97, 32
40576 DATA 110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40592 DATA  32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40608 DATA  32, 32, 32, 32, 32, 32, 97, 32,110, 32, 32, 32, 32, 32, 32, 32
40624 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
40640 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40656 DATA 110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40672 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 23,  3, 32, 32
40688 DATA  32, 32, 32, 32, 32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32
40704 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32, 32, 32, 32, 32
40720 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40736 DATA 110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40752 DATA  32, 32, 32, 32,110, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40768 DATA  32, 32, 32, 32, 32, 32, 97, 32,110, 32, 32, 32, 32, 32, 32, 32
40784 DATA  32, 32, 32, 32, 32, 32, 97, 32, 32, 32, 32, 32,110, 32, 32, 32
40800 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 97, 32
40816 DATA  32,120,120,120,120,120,120,120,120,120,120,120,120,120,120,120
40832 DATA 120,120,120,120,120,120,120,120,120,120,120,120,120,120,120,120
40848 DATA 120,120,120,120,120,120,126, 32, 32, 32, 32, 32, 32, 32, 32, 32
40864 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
40880 DATA  32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32, 32
```

I Data che codificano il disegno della foto in alto a sinistra

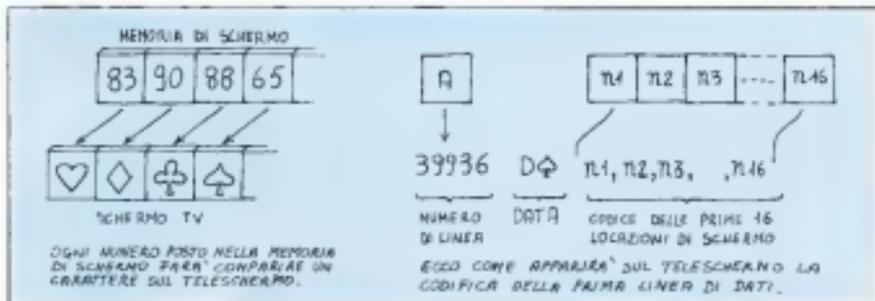
linee Data il contenuto del pezzo di memoria che va dalla locazione 39936 in su per 960 byte naturalmente potrebbe trattarsi di qualsiasi altra zona di memoria. La prima cosa importante che viene fatta (1020) e quella di azzerare una locazione di Ram non utilizzata dal sistema operativo per memorizzarvi una variabile che ci farà comodo nel corso dell'elaborazione Nella linea 1040 viene definita una variabile

A = 39936 + 16*PEEK(252)

che verrà poi incrementata di 16 unità per volta incrementando di un'unità il contenuto della locazione 252 (1090) In altre parole, quando PEEK (252) = 0 (contenuto della locazione 252), allora A = 39936, quando PEEK (252) = 1 segue che A vale 39952 e così via (se ancora non capite non preoccupatevi perchè tra breve metteremo insieme tutti gli elementi) Con la linea 1050 viene scritta sullo schermo la stringa di due caratteri che compare tra virgolette e che rappresenta l'abbreviazione della parola chiave "DATA" mentre dalla linea 1100 a 1130 che sintetizziamo nella seguente tabella

Loc.	Cod	Carattere
631	145	curs au
632	145	curs au
633	13	Return
634	71	G
635	207	Shift + 0
636	49	1
637	49	1
638	52	4
639	48	0
640	13	Return

La sequenza rappresentata, come è facile osservare, oltre a spostare il cursore due posizioni verso l'alto genera anche un GOTO 1140 Ciò viene fatto per una ragione che abbiamo discusso la volta scorsa ma su cui vogliamo ritornare per maggiore chiarezza. Dato che il Buffer della tastiera viene svuotato solo dopo un "END" (lo trosioni del Basic 1K di memoria posto nella parte più alta di quella disponibile per il programma (linea 3) La linea 6 cambia il colore dello schermo e del carattere; la 10 mette il programma in attesa oltre ad inviare, ad intervalli regolari, ad una subroutine (100-150) che simula il lampeggio del cursore, cosa che sfrutteremo per orientarci sullo schermo nel corso della composizione del disegno Vediamo ora le opzioni delle linee 30, 50 e 60.

Premendo il tasto f1 il contenuto dello schermo viene copiato nella zona precedentemente protetta che comincia dalla locazione 39936 (156*256). Questo perchè durante il processo di autoDATA, per quanto detto precedentemente, il contenuto della memoria schermo vera e propria andrebbe perduto Premendo f3 è invece possibile ricostruire sul monitor il contenuto dello schermo dopo la precedente operazione di copiatura. Come è immediato constatare, il contenuto della zona protetta è scaricato in memoria video e viene

1060 alla 1080 viene composta una stringa, A$, contenente i numeri situati nelle prime 16 locazioni da codificare, separati da una virgola

In base a questi primi elementi, vediamo come si comporta il programma fino alla linea 1090.

Azzerata la locazione 252 e cancellato lo schermo viene calcolato il primo valore di A, poi vengono scritte sullo schermo A, D + <il simbolo di picche>, A$ e viene incrementata la locazione 252 In definitiva a questo punto sullo schermo vedremo scritto

39936 DATA n1 n2, n16

dove n1 n16 sono i contenuti delle prime 16 locazioni interessate Specifichiamo che la parola chiave DATA verrà scritta in forma abbreviata (D + picche) ma noi la riportiamo per intero per esigenze tipografiche

A questo punto ci vuole una opportuna sequenza scritta nel Buffer di tastiera che (come già sapete) porti il cursore sulla linea scritta e generi un Return affinchè essa sia introdotta nel programma. Ciò è ottenuto con la sequenza contenuta nelle linee da vate in linea 1130 dopo POKE 198,10 che indica il numero di caratteri che devono essere scaricati sullo schermo), questa necessità produrrebbe un arresto del programma e noi vogliamo che ciò si verifichi al fine di rendere automatica tutta la procedura Introduciamo allora l'invio alla linea 1140 che fa ripartire il programma da tale posizione

Capite ora l'importanza di aver conservato il valore del contatore del numero di linea Data in una locazione Ram con una Poke e non come una variabile nel modo consueto con un Goto, tale variabile verrebbe perduta

Tutto il procedimento viene ripetuto per un numero di volte tale da coprire tutta l'area interessata

Vogliamo farvi notare che un tale sistema di conversione potrebbe essere utilizzato anche per introdurre in linee Data un programma in Lm per poi utilizzarlo in un caricatore Basic

Analizzata la parte più importante del programma passiamo ad illustrare le parti principali di ciò che rimane

Per prima cosa viene protetto da intrusioni riempita di "1" la memoria del colore Vi facciamo osservare che abbiamo preferito non implementare la codifica dell'ultima linea video che non potrebbe essere riprodotta fedelmente sia per la presenza su di essa del cursore sia per la necessità di lasciare l'ultima locazione di tale riga in bianco onde evitare lo scroll verso l'alto a causa dell'"a capo" generato impegnando l'ultima locazione del video Infine il tasto f5 avvia il procedimento di codifica che già conosciamo.

Come isolare le linee di dati

Una volta ottenute le linee di programma con i Data dovremo in qualche modo isolarle ed introdurle in un contesto che le renda utilizzabili La prima cosa che ci viene da pensare è — partendo dal programma già elaborato contenente le linee aggiuntive — di cancellare le linee che precedono la 39936 (la prima linea di dati) scrivendo semplicemente i numeri delle righe che vogliamo cancellare seguiti da Return. Questo primo procedimento non ci lascia comunque molto soddisfatti certo

tutto sarebbe molto più facile se avessimo un comando di "Delete" come, ad esempio, il C 16 o il Plus 4, ma purtroppo un tale comando non c'è e quindi dobbiamo arrangiarci. Pensando un po' ci viene in mente che un metodo per rendere "invisibili" al sistema operativo un certo numero di linee esiste ed è abbastanza semplice da mettere in atto

Oramai dovreste sapere tutto (se non altro perché ne abbiamo ampiamente parlato in questa stessa rubrica) su alcuni dei principali puntatori del sistema e su come viene formattato un programma in memoria. Riferendoci — sempre per fissare le idee — al 64, l'area utilizzata per i programmi inizia dalla locazione 2049 (in effetti dalla 2048 che contiene il flag 0) ed è identificabile leggendo il valore di un puntatore a due byte contenuto nelle locazioni 43 e 44 nelle quali, all'accensione, troveremo i numeri 1 e 8. Da una rapida verifica ricaviamo infatti che

```
1+8*256=2049
```

Il sistema comincia il suo lavoro sul programma a partire dal punto indicato dal contenuto di queste due locazioni. In memoria, ogni linea rappresenta un blocco a se il quale comprende dei byte che permettono al sistema operativo la concatenazione dei vari blocchi. Ciascuno di essi inoltre è separato dal successivo per mezzo di uno "0" ed infine il programma termina ponendo la serie "000" in calce all'ultimo dei blocchi in questione. La fine del programma viene puntata dal contenuto delle locazioni 45 e 46

Se noi, avendo un programma in memoria, manipoliamo opportunamente i puntatori descritti, potremo ingannare facilmente il sistema: sfrutteremo questo fatto per i nostri scopi. Cominciamo le nostre osservazioni e modifiche partendo dal programma di autoDATA

Dopo aver caricato in memoria tale programma, proviamo a leggere il contenuto dei puntatori di fine programma, otterremo

```
PEEK (45) = 240 e PEEK (46) = 11
```

cioè viene puntata la locazione 3056. La prima osservazione da fare è che il programma termina praticamente se comprendiamo lo zero di fine blocco, alla locazione 3053 (compresa) e gli altri due byte aggiuntivi si riferiscono ai due zero che, insieme a quello precedente, realizzano la sequenza "000" cui accennavamo poc'anzi. Appuntiamoci i valori 240, 11 e 3053 e andiamo avanti. Disegnamo ciò che vogliamo sullo schermo e avviamo la sequenza di autoDATA: per quanto detto le linee aggiuntive verranno aggiunte a partire dalla locazione 3053 ed i due zero verranno spostati alle fine della codifica dell'ultimo blocco di tali linee. Naturalmente tutto ciò avviene in modo a noi del tutto trasparente. A questo punto, se vogliamo far ignorare al sistema tutto ciò che precede le linee con i Data, basterà comunicargli che il programma non inizia più dalla locazione 2049 ma dalla 3053 e questo è semplicemente ottenibile con:

```
POKE43 238 POKE44 11
```

Se ora date il List, vedrete che non compariranno più le linee precedenti la 39936 e quindi avremo isolato tutte quelle con i Data. Per mettere in funzione autonomamente ciò che avete ricavato, provate a questo punto ad aggiungere il seguente segmento

```
10 FORI=55296 TO 56295
20 POKE I,1 NEXT
30 FORI=0TO959 READ A
40 POKE1024+I,A NEXT
50 GOTO 50
```

e a dare il Run: vedrete il vostro disegno ricomparire sullo schermo.

Un semplice APPEND

È evidente che potreste impiegare le linee appena isolate in un altro programma e sarebbe scomodo doverle ribattere tutte in macchina. Se avessimo un comando di Merge o di Append potremmo ottenere lo scopo semplicemente ma, dato che non ne disponiamo, dovremo anche questa volta inventarci qualche cosa. Di seguito vi proponiamo una sequenza che permette di appendere un programma in coda ad un altro: l'unica condizione perché ciò possa verificarsi è che i due programmi non devono avere linee etichettate nello stesso modo ed inoltre il massimo numero di linea del primo programma deve essere inferiore al minimo numero del secondo. Nel nostro caso questa condizione si verifica abbastanza naturalmente essendo le linee numerate da 39936 in poi, negli altri casi, se siamo noi a scrivere i programmi da unire l'uno con l'altro, faremo attenzione a numerarli in maniera opportuna. La sequenza proposta è la seguente

```
1) A= PEEK(43)
2) B= PEEK(44)
3) C= 256*PEEK(46)+PEEK(45)-2
4) POKE43 C AND 255
5) POKE44 INT(C/256) NEW
6) Caricara il programma da appendara
7) POKE43 A POKE44 B CLR
```

Essa non ha bisogno di molti commenti: A e B per il C64 sono 1 e 8, con il terzo passaggio viene ricavato il punto da cui cominciare l'Append mentre con il quarto ed il quinto tale punto viene scomposto nei due valori da inserire nel puntatore di inizio programma ed azzerato quello di fine programma. Ciò permette al programma che sarà caricato di non andarsi a sovrapporre a quello già in memoria. Dopo il caricamento viene riabbassato il puntatore 43/44 scoprendo così entrambi i programmi. Il procedimento descritto può essere ripetuto più volte per appendere programmi fino a riempire tutta l'area di memoria disponibile

Per il Vic

Tutti i discorsi fatti finora per il C 64 restano concettualmente gli stessi per il Vic 20. Naturalmente bisogna tener presente alcune differenze tra l'una e l'altra macchina tra cui le più rilevanti sono il diverso formato di schermo e la mappa della memoria che varia a seconda dell'espansione posseduta. Naturalmente, chi ci ha sempre seguiti troverà facilmente le modifiche da effettuare, ma anche gli altri sicuramente non troveranno molte difficoltà

Sperando di aver stimolato in qualche modo la vostra fantasia vi diamo appuntamento al prossimo numero sempre ricco di interessanti informazioni sul Vic e sul C 64

L'ADP BASIC

gli ultimi 6

di Andrea De Prisco

Si conclude con questo numero la presentazione del tool device-oriented ADP Basic. Con quest'ultima porzione di interprete potremo facilmente gestire file relativi, definire gli 8 tasti F1-F8 e leggere un file sequenziale visualizzando il suo contenuto su video, alla maniera del comando VIEW, valido solo per i programmi Basic.

Quinta parte

Per chi non ha gli ultimi 4 numeri di MCmicrocomputer o, più semplicemente, per tutti coloro che non sono interessati a come e perché l'ADP Basic funziona, volendo soltanto mandarlo in esecuzione e usarlo, in queste pagine è stato ripubblicato l'intero listato dei 4096 byte occupati dal Tool ADP. Un apposito riquadro, inoltre, elenca una per una tutte le istruzioni implementate, non senza indicare le opportune avvertenze e modalità d'uso.

Chi ha la stampante 801 (e, ripetiamo, non ci ha seguito nei numeri scorsi) dopo aver digitato il listato 3, inserisca in memoria anche il listato 4 che adatterà l'ADP Basic alla stampante minore.

I 6 nuovi comandi

La prima istruzione presentata questo mese permette di definire i tasti F1-F8. Ad ognuno è possibile associare una stringa di massimo 8 caratteri. Il comando è KEY, e ha due diversi significati. Se non è seguito da nulla, attiva le 8 definizioni e le visualizza una per una sullo schermo.

```
10 INPUT"NOME FILE REL.";A$
20 N=0:INPUT"LUNGHEZZA REG.";N
30 IFN=0THEN100
40 RELOPEN A$,N
50 N=0:INPUT"N. REGISTRAZIONE";N
60 IFN<=0THEN150
70 INPUT"REGISTRAZIONE    ";A$
80 SEND N,A$
90 GOTO50
100 RELOPEN A$
110 N=0:INPUT"N. REGISTRAZIONE";N
120 IFN<=0THEN150
130 RECEIVE N,A$:PRINTA$
140 GOTO110
150 RELCLOSE
READY.
```

Listato Basic 1 - Esempio di trattamento di file relativi

```
284 FORI=49790TO50175:READI1:POKEI,I1:NEXT
285 FORI=52892TO53186:READI1:POKEI,I1:NEXT
286 FORI=49374TO49391:READI1:POKEI,I1:NEXT
287 FORI=52946TO52983:READI1:POKEI,I1:NEXT
288 POKE51758,0:POKE51787,207
300 ++++++++++++++++++++++++++++++++++++++++++++++++++++++
310 +                                                    +
320 +   **4  **4  ***       ***   ***   ***  **4  ***    +
330 +  *   * *  * *   *     *  * *  * 4 *       *  *  *  +
340 +  ****4 *   * 4444     4*4* 4444 4*4    *  *        +
350 +  *   * *  * *         *  * *  *     4  *  *   4    +
360 +  *   * ****  *        ****  4   * 4**  ***  ***    +
370 +                                                    +
380 +           -------------------------                +
390 +           (C)  1984      ADP SOFTWARE              +
400 +           -------------------------                +
410 +                                                    +
500 ++++++++++++++++++++++++++++++++++++++++++++++++++++++
6004 DATA120,189,120,141,143,2,169,185,141,144,2,88,32,0,182,32,121,0,201,0,240
6014 DATA34,201,58,240,80,234,234,234,201,49,144,80,201,57,176,70,41,15,170,202
6024 DATA138,72,32,115,0,201,44,228,83,32,115,0,32,158,173,36,13,16,53,32,183
6034 DATA182,133,80,104,170,189,184,195,10,10,10,170,160,0,177,34,240,18,201
6044 DATA31,208,2,128,13,157,182,195,232,200,192,8,240,15,186,80,208,233,189
6054 DATA0,157,182,195,232,208,192,8,208,247,78,248,188,76,43,188,182,0,169,13
6064 DATA32,210,255,188,75,32,210,255,189,69,32,210,255,189,85,32,210,255,138
6074 DATA24,105,1,5,48,32,210,255,165,44,32,210,255,169,34,32,210,255,168,184
6084 DATA195,10,10,10,168,138,72,182,0,185,192,195,240,0,201,13,208,2,169,31
6094 DATA32,210,255,200,232,224,9,208,238,189,34,32,210,255,104,170,232,201,7
6104 DATA208,174,76,248,168,32,212,225,189,183,240,159,168,85,182,8,160,4,32
6114 DATA188,255,32,192,255,162,85,32,198,255,32,207,255,32,210,255,32,225,255
6124 DATA240,5,32,183,255 240,240,32,204,255,189 65,32 195,255,76,140,186,0,0
6134 DATA0,0,0,0,0,0,185,203,197,197,208,3,79,72,235,201,3,144,248,001,7,170
6144 DATA245,56,233,3,10,10,10,172,141,2,240,3,24,185,32,162,7,24,153,7,169,183
6154 DATA182,185,197,110,2,136,202,16,249,189,0,133,198,79,72,235,0,0,1,5,2,0
6164 DATA3,7,0,4,87,65,84,13,0,0,0,0,70,78,65,83,72,13,0,0,88,83,65,88,89,32
6174 DATA0,0,68,76,78,85,68,32,0,0,69,82,65,83,82,32,0,0,80,80,69,34,58,42,34
6184 DATA13,85,80,88,88,78,80,0,0 88,73,83,87,32,32,0,0
8200 REM ***************************************************************
8204 DATA201,159,240,7,201,198,248,87,79,43,199,32,59,182,32,115,0,32,212,229
8214 DATA164,183,240,240,138,177,187,183,84,3,182,0,208,240,184,183,189,180,201
8224 DATA1,240,24,188,44,153,84,3,200,189,78,153,84,3,200,189,44,153,84,3,208
8234 DATA165,188 153,84,3,208,152,182,84,168,3,32,185,255,169,66,168,0,180,4
8244 DATA32,186,255,32,182,255,182,72,32,188,255,32,207,255,32,183,255,201,84
8254 DATA228 246,32,204,255,88,32,204,255,169,80,32,195,255,78,240,168,32,138
8264 DATA173,32,247,183,128,72,32,201,255,189,80,32,210,255,189,4,32,210,255
8274 DATA165,20,32,210,255,165,21,32,210,255,169,1,32,210,255,32,204,255,78,80
8284 DATA207,234,234,234,32,119,207 32,253,174,152,86,32,201,255,189,13,32,180
8294 DATA178,32,204,255,78,240,168,32,118,207,32,253,174,182,86,78,173,171
8300 REM ***************************************************************
8394 DATA121,184,180,207,128,207,68,155
8420 REM ***************************************************************
8484 DATA75,65,217,83,80,78,188,68,59,87,68,73,80,187,78,73,76,187
READY.
```

Listato Basic 2 - Linee da aggiungere a quelle dei numeri scorsi per implementare le istruzioni di questo mese

Questa operazione è inoltre necessaria, ogni qual volta si esegue la sequenza Run/Stop e Restore, che modifica alcune celle in pagina zero. Se nessuna modifica è fatta alle definizioni degli otto tasti, troveremo al tasto F1 il comando FLASH, al tasto F3 il comando DSAVE, al tasto F5 il comando DLOAD, al tasto F7 il comando CAT per visualizzare la directory del dischetto presente nel Driver.

Se facciamo precedere la pressione dallo Shift, troveremo rispettivamente le sequenze: EXE "*" per caricare il primo programma da dischetto, APPEND per saldare due programmi, VIEW per mostrare su

204	INIT	217	BSAVE	230	MOVE	243	PAGE
205	VDATE	218	RANGE	231	DRAW	244	FEED
206	ERASE	219	TRSE	232	REL	245	MESSAGE
207	FMAT	220	APPEND	233	COLOR	246	LPRINT
208	COPY	221	DISKNAME	234	BLACK	247	USING
209	RENAME	222	VIEW	235	BLUE	248	OUT
210	FLASH	223	RESET	236	GREEN	249	HCOPY
211	DLOAD	224	OFF	237	RED	250	LLIST
212	DSAVE	225	SMALL	238	SIZE	251	KEY
213	OVER	226	CAP	239	ROTATE	252	SEND
214	CAT	227	PL	240	BROKEN	253	RECEIVE
215	EXE	228	HOME	241	WRITE	254	FILE
216	BLOAD	229	SET	242	PR		

Figura 1 - Tutte le istruzioni dell'ADP Basic, precedute dal codice TOKEN

```
C27A 78       SEI
C27B A9 80    LDA #$80
C27D 8D 8F 02 STA $028F
C280 A9 C3    LDA #$C3
C282 8D 90 02 STA $0290
C285 58       CLI
C286 20 00 C0 JSR $C000
C289 20 79 00 JSR $0079
C28C C9 00    CMP #$00
C28E F0 5E    BEQ $C2EE
C290 C9 3A    CMP #$3A
C292 F0 5A    BEQ $C2EE
C294 EA       NOP
C295 EA       NOP
C296 EA       NOP
C297 C9 31    CMP #$31
C299 90 50    BCC $C2EB
C29B C9 39    CMP #$39
C29D B0 4C    BCS $C2EB
C29F 29 0F    AND #$0F
C2A1 AA       TAX
C2A2 CA       DEX
C2A3 8A       TXA
C2A4 48       PHA
C2A5 20 73 00 JSR $0073
C2A8 C9 2C    CMP #$2C
C2AA D0 3F    BNE $C2EB
C2AC 20 73 00 JSR $0073
C2AF 20 9E AD JSR $AD9E
C2B2 24 0D    BIT $0D
C2B4 10 35    BPL $C2EB
C2B6 20 A3 B6 JSR $B6A3
C2B9 85 50    STA $50
C2BB 68       PLA
C2BC AA       TAX
C2BD BD B8 C3 LDA $C3B8,X
C2C0 0A       ASL
C2C1 0A       ASL
C2C2 0A       ASL
C2C3 AA       TAX
C2C4 A0 00    LDY #$00
C2C6 B1 22    LDA ($22),Y
C2C8 F0 13    BEQ $C2DD
C2CA C9 1F    CMP #$1F
C2CC D0 02    BNE $C2D0
C2CE A9 0D    LDA #$0D
C2D0 9D C0 C3 STA $C3C0,X
C2D3 E8       INX
C2D4 C8       INY
C2D5 C0 08    CPY #$08
C2D7 F0 0F    BEQ $C2E8
C2D9 C4 50    CPY $50
C2DB D0 E9    BNE $C2C6
C2DD A9 00    LDA #$00
C2DF 9D C0 C3 STA $C3C0,X
C2E2 E8       INX
C2E3 C8       INY
C2E4 C0 08    CPY #$08
C2E6 D0 F7    BNE $C2DF
C2E8 4C F8 A8 JMP $A8F8
C2EB 4C 28 C6 JMP $C628
C2EE A2 00    LDX #$00
C2F0 A9 0D    LDA #$0D
C2F2 20 D2 FF JSR $FFD2
C2F5 A9 4B    LDA #$4B
C2F7 20 D2 FF JSR $FFD2
C2FA A9 45    LDA #$45
C2FC 20 D2 FF JSR $FFD2
C2FF A9 59    LDA #$59
C301 20 D2 FF JSR $FFD2
C304 8A       TXA
C305 18       CLC
C306 69 01    ADC #$01
C308 09 30    ORA #$30
C30A 20 D2 FF JSR $FFD2
C30D A9 2C    LDA #$2C
C30F 20 D2 FF JSR $FFD2
C312 A9 22    LDA #$22
C314 20 D2 FF JSR $FFD2
C317 BD B8 C3 LDA $C3B8,X
C31A 0A       ASL
C31B 0A       ASL
C31C 0A       ASL
C31D A8       TAY
C31E 8A       TXA
C31F 48       PHA
C320 A2 00    LDX #$00
C322 B9 C0 C3 LDA $C3C0,Y
C325 F0 09    BEQ $C330
C327 C9 0D    CMP #$0D
C329 D0 02    BNE $C32D
C32B A9 1F    LDA #$1F
C32D 20 D2 FF JSR $FFD2
C330 C8       INY
C331 E8       INX
C332 E0 08    CPX #$08
C334 D0 EC    BNE $C322
C336 A9 22    LDA #$22
C338 20 D2 FF JSR $FFD2
C33B 68       PLA
C33C AA       TAX
C33D E8       INX
C33E C9 07    CMP #$07
C340 D0 AE    BNE $C2F0
C342 4C F8 A8 JMP $A8F8
```

Listato 1 - Istruzione KEY

```
C345 20 D4 E1 JSR $E1D4
C348 A5 B7    LDA $B7
C34A F0 9F    BEQ $C2EB
C34C A9 41    LDA #$41
C34E A2 08    LDX #$08
C350 A0 04    LDY #$04
C352 20 BA FF JSR $FFBA
C355 20 C0 FF JSR $FFC0
C358 A2 41    LDX #$41
C35A 20 C6 FF JSR $FFC6
C35D 20 CF FF JSR $FFCF
C360 20 D2 FF JSR $FFD2
C363 20 E1 FF JSR $FFE1
C366 F0 05    BEQ $C36D
C368 20 B7 FF JSR $FFB7
C36B F0 F0    BEQ $C35D
C36D 20 CC FF JSR $FFCC
C370 A9 41    LDA #$41
C372 20 C3 FF JSR $FFC3
C375 4C 8C C4 JMP $C48C
```

Listato 2 - Istruzione FILE

```
C380 A5 CB    LDA $CB
C382 C5 C5    CMP $C5
C384 D0 03    BNE $C389
C386 4C 48 EB JMP $EB48
C389 C9 03    CMP #$03
C38B 90 F9    BCC $C386
C38D C9 07    CMP #$07
C38F B0 F5    BCS $C386
C391 38       SEC
C392 E9 03    SBC #$03
C394 0A       ASL
C395 0A       ASL
C396 0A       ASL
C397 AC 8D 02 LDY $028D
C39A F0 03    BEQ $C39F
C39C 18       CLC
C39D 69 20    ADC #$20
C39F A2 07    LDX #$07
C3A1 18       CLC
C3A2 69 07    ADC #$07
C3A4 A8       TAY
C3A5 B9 C0 C3 LDA $C3C0,Y
C3A8 9D 77 02 STA $0277,X
C3AB 88       DEY
C3AC CA       DEX
C3AD 10 F6    BPL $C3A5
C3AF A9 08    LDA #$08
C3B1 85 C6    STA $C6
C3B3 4C 48 EB JMP $EB48
```

Listato 3 - Modifica alla routine di scansione della tastiera

```
CF00 C9 9F    CMP #$9F
CF02 F0 07    BEQ $CF0B
CF04 C9 A0    CMP #$A0
CF06 F0 01    BEQ $CF09
CF08 4C 28 C6 JMP $C628
CF0B 20 38 C0 JSR $C038
CF0E 20 73 00 JSR $0073
CF11 20 D4 E1 JSR $E1D4
CF14 A4 B7    LDY $B7
CF16 F0 F0    BEQ $CF08
CF18 88       DEY
CF19 B1 BB    LDA ($BB),Y
CF1B 99 40 03 STA $0340,Y
CF1E C0 00    CPY #$00
CF20 D0 F6    BNE $CF18
CF22 A4 B7    LDY $B7
CF24 A5 BA    LDA $BA
CF26 C9 01    CMP #$01
CF28 F0 18    BEQ $CF42
CF2A A9 2C    LDA #$2C
CF2C 99 40 03 STA $0340,Y
CF2F C8       INY
CF30 A9 4C    LDA #$4C
CF32 99 40 03 STA $0340,Y
CF35 C8       INY
CF36 A9 2C    LDA #$2C
CF38 99 40 03 STA $0340,Y
CF3B C8       INY
CF3C A5 BA    LDA $BA
CF3E 99 40 03 STA $0340,Y
CF41 C8       INY
CF42 98       TYA
CF43 A2 40    LDX #$40
CF45 A0 03    LDY #$03
CF47 20 BD FF JSR $FFBD
CF4A A9 42    LDA #$42
CF4C A2 08    LDX #$08
CF4E A0 04    LDY #$04
CF50 20 BA FF JSR $FFBA
CF53 20 C0 FF JSR $FFC0
CF56 A2 48    LDX #$48
CF58 20 C6 FF JSR $FFC6
CF5B 20 CF FF JSR $FFCF
CF5E 20 B7 FF JSR $FFB7
CF61 C9 40    CMP #$40
CF63 D0 F6    BNE $CF5B
CF65 20 CC FF JSR $FFCC
CF68 60       RTS
CF69 20 CC FF JSR $FFCC
CF6C A9 42    LDA #$42
CF6E 20 C3 FF JSR $FFC3
CF71 4C F8 A8 JMP $A8F8
```

Listato 4 - Istruzioni REL CLOSE e REL OPEN

```
CF74 20 8A AD JSR $AD8A
CF77 20 F7 B7 JSR $B7F7
CF7A A2 48    LDX #$48
CF7C 20 C9 FF JSR $FFC9
CF7F A9 50    LDA #$50
CF81 20 D2 FF JSR $FFD2
CF84 A9 04    LDA #$04
CF86 20 D2 FF JSR $FFD2
CF89 A5 14    LDA $14
CF8B 20 D2 FF JSR $FFD2
CF8E A5 15    LDA $15
CF90 20 D2 FF JSR $FFD2
CF93 A9 01    LDA #$01
CF95 20 D2 FF JSR $FFD2
CF98 20 CC FF JSR $FFCC
CF9B 4C 56 CF JMP $CF56
```

Listato 5 - Questa routine usata da SEND e RECEIVE serve per posizionarsi su una registrazione del file relativo

```
CFA1 20 74 CF JSR $CF74
CFA4 20 FD AE JSR $AEFD
CFA7 A2 42    LDX #$42
CFA9 20 C9 FF JSR $FFC9
CFAC A9 0D    LDA #$0D
CFAE 20 A0 AA JSR $AAA0
CFB1 20 CC FF JSR $FFCC
CFB4 4C F8 A8 JMP $A8F8
```

Listato 6 - Istruzione SEND

```
CFB7 20 74 CF JSR $CF74
CFBA 20 FD AE JSR $AEFD
CFBD A2 42    LDX #$42
CFBF 4C AD AB JMP $ABAD
```

Listato 7 - Istruzione RECEIVE

video il listato di un programma che sta su disco, ERASE per cancellare file inutili dal dischetto

Le definizioni relative ai tasti F1, F2 e F7, forzano automaticamente un [RETURN], per eseguire la funzione con la sola pressione del tasto definibile

Per cambiare la definizione dei tasti, é sufficiente digitare

KEY NumeroTasto "Stringa"

eventualmente seguito da un +CHR$(13) se si desidera un'auto-esecuzione dell'istruzione contenuta nella stringa Facciamo un esempio, vogliamo associare al tasto F1 la stringa CIAO Digiteremo semplicemente

KEY 1,"CIAO"

da questo momento, ogni pressione del tasto F1, fara apparire su video la sequenza di caratteri C, I, A, O Se vogliamo definire un tasto con un comando, ad esempio LIST, digiteremo

KEY 1,"LIST"+CHR$(13)

ad ogni pressione del tasto F1, sarà eseguito il list su video del programma Basic contenuto in memoria

Per visualizzare il contenuto di un file sequenziale, alla maniera del comando VIEW valido solo per i programmi Basic, é disponibile il comando FILE, seguito dal nome del file

È possibile anche leggere file dell'EASY SCRIPT, precedentemente salvati dal word processor Se non si desidera leggere tutto il file, é sufficiente agire sul tasto Run/Stop per interrompere anzitempo la visualizzazione

Per un trattamento facilitato dei file relativi, sono disponibili 4 nuovi comandi

REL OPEN "NomeFile",LunghezzaReg

Permette di inizializzare un file relativo i cui elementi hanno lunghezza minore o uguale a quella indicata dopo la virgola (compresa sempre tra 2 e 80) L'operazione di inizializzazione e obbligatoria prima di cominciare a inserire elementi Per aprire semplicemente un file relativo, gia inizializzato, per successive operazioni di lettura e/o scrittura, basta non specificare la lunghezza degli elementi

Quando non si usa piú un file relativo, bisogna chiuderlo col comando REL CLOSE, seguito da nulla

Per inserire registrazioni nel file é disponibile il comando SEND (invia) La sua sintassi é.

SEND Posizione,"Stringa"

e inserisce la "Stringa" (che non deve contenere virgole o [Return]) nel file relativo aperto, nella posizione specificata Per recuperare una registrazione si usa il comando RECEIVE, seguito dalla posizione che ci interessa e da una variabile di tipo stringa che, a operazione compiuta, conterrá la registrazione cercata. Come per il coman-

Questo programma è disponibile su disco presso la redazione Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag 157.

```
10 FORI=49152TO53247:READII:POKEI,II:NEXT:SYS49152:END
20 ************************************************
30 *                                              *
35 *   ***  ****  ****     ****   ***   ***  ***   *
40 *  *   * *   * *   *   *   * *   * *     *  *   *
45 *  ***** *   * ****    ****  ***** ***   *  *   *
50 *  *   * *   * *       *   * *   *    *  *  *   *
55 *  *   * ****  *       ****  *   * ***  *** ***  *
60 *                                              *
65 *          ------ -------------- -             *
68 *         (C) 1984     ADP SOFTWARE            *
70 *          ------------------------            *
75 *                                              *
80 ************************************************
100 DATA120,169,32,141,4,3,169,183,141,5,3,169,88,141,8,3,125,184,141,7,3,200
110 DATA234,168,200,141,8,3,168,152,141,8,3,200,48,32,204,255,32,201,205,168
120 DATA88,160,4,168,8,32,186,255,132,8,32,185,255,32,192,206,88,32,000,255,32
130 DATA231,255,168,72,162,8,160,15,32,186,255,160,8,32,189,255,32,192,255,88
140 DATA169,11,141,32,200,141,33,220,169,155,32,210,255,88,169,187,141,7,3,141
150 DATA8,3,169,185,141,5,3,159,124,141,4,3,169,25,141,8,3,125,228,141,8,3,78
160 DATA248,168,255,125,18,168,37,168,82,168,81,168,168,168,138,168,171,168,183
170 DATA166,125,186,18,187,181,187,200,187,208,187,171,168,248,168,164,168,186
180 DATA128,128,208,225,252,38,152,42,168,42,168,137,201,236,201,251,201,4,202
190 DATA22,202,31,202,37,202,118,202,135,202,130,000,143,208,147,208,187,208
200 DATA238,202,240,202,38,203,122,203,188,203,203,203,236,203,3,204,24,206,82
210 DATA204,213,204,121,194,168,207,168,207,88,155,0,0,0,0,0,0,0,0,0,0,0,0,0,0
220 DATA82,0,0,0,0,0,32,115,0,201,128,48,4,201,208,15,4,88,76,231,167,52,233,000
230 DATA10,166,168,187,72,185,233,72,185,128,162,72,125,128,182,72,78,115,0,188
240 DATA122,166,4,132,15,188,8,2,16,7,201,255,248,73,232,000,244,001,32,248,88
250 DATA133,8,201,34,248,57,56,13,112,82,201,63,208,8,185,153,208,46,201,48,144
260 DATA4,201,88,144,46,132,113,168,8,169,208,133,254,168,1,133,253,132,11,166
270 DATA134,122,208,208,208,2,232,254,232,169,8,2,56,241,203,248,242,201,168
280 DATA208,48,5,11,164,113,232,200,183,231,1,185,251,1,248,88,82,233,82,248
290 DATA4,201,73,208,2,133,15,56,233,25,208,148,133,8,188,8,2,248,223,187,6,248
300 DATA215,208,153,251,1,232,208,248,168,122,238,11,168,253,208,208,2,238,208
310 DATA177,253,15,247,232,053,177,203,208,175,185,8,2,16,184,153,253,1,188,123
320 DATA168,235,133,128,25,144,8,248,4,201,171,209,247,32,187,168,32,18,185,32
330 DATA121,8,248,12,201,171,208,232,32,115,8,32,187,169,208,224,184,184,125
340 DATA28,5,21,208,6,189,255,133,28,133,21,168,1,132,15,177,85,248,87,32,44
350 DATA182,32,215,178,208,177,25,178,208,177,85,187,21,208,4,228,28,248,2,176
360 DATA44,132,73,32,225,168,188,32,164,73,41,127,32,71,171,201,34,208,8,188
370 DATA13,73,255,133,15,208,248,17,177,85,208,18,168,177,83,178,208,177,85,134
380 DATA55,133,88,208,161,78,134,227,168,8,3,16,215,201,255,240,211,38,15,48
390 DATA207,56,233,127,170,132,73,186,255,169,204,133,255,169,1,133,254,000,202
400 DATA11,208,206,2,255,253,177,254,18,247,48,242,208,000,2,255,255,177,254
410 DATA48,184,32,71,171,000,242,128,188,128,141,143,2,188,185,141,144,2,88,32
420 DATA82,192,32,121,8,201,8,248,94,201,58,248,88,234,234,234,201,48,144,88,201
430 DATA87,178,76,41,16,170,202,132,72,32,115,0,201,44,208,82,32,115,8,32,188
440 DATA173,38,13,16,53,32,169,182,133,88,184,176,168,184,185,10,10,10,178,188
450 DATA8,177,38,248,18,201,31,255,2,169,13,157,182,188,232,208,182,8,248,15
460 DATA188,88,208,233,169,8,157,182,125,232,208,182,8,254,247,78,248,169,78
470 DATA43,188,162,8,132,13,32,218,255,169,75,32,218,255,168,63,32,218,255,188
480 DATA88,32,218,255,132,24,185,1,5,48,32,218,255,168,44,32,218,255,188,34,32
490 DATA218,255,188,184,125,18,18,18,182,138,72,182,8,169,192,185,848,8,261,13
500 DATA202,2,188,31,32,218,255,208,232,224,8,208,230,169,34,32,218,255,184,178
510 DATA232,201,7,208,178,78,248,188,32,212,225,182,143,248,132,188,85,188,8
520 DATA162,4,32,128,255,32,188,255,182,65,32,188,255,32,287,255,32,218,255,32
530 DATA225,255,208,5,32,183,255,208,240,32,204,255,169,85,32,189,205,76,182
540 DATA188,0,0,0,0,0,0,0,0,185,203,197,187,202,3,78,72,235,201,3,144,248,201
550 DATA7,175,245,88,233,3,18,16,18,178,141,2,248,3,84,185,32,188,7,84,185,7
560 DATA168,185,168,185,157,115,2,138,208,18,246,185,8,123,188,78,78,235,0,0
570 DATA1,5,2,0,3,7,0,0,67,65,84,13,0,0,0,0,78,78,85,83,72,13,0,0,88,85,69,88
580 DATA65,32,0,0,88,76,78,88,88,32,0,0,88,82,65,83,88,32,0,0,88,88,88,34,38
590 DATA42,64,13,65,88,88,88,78,82,0,0,88,73,69,87,32,32,0,0,32,88,182,162,72
600 DATA32,201,205,188,73,32,218,255,32,231,255,78,248,169,32,88,182,188,72,32
610 DATA201,255,182,88,32,218,255,32,231,255,78,248,168,32,58,182,188,78,32,201
620 DATA255,185,83,32,218,255,208,87,32,52,182,188,72,32,201,253,188,78,32,218
630 DATA235,158,58,32,218,255,32,212,225,188,8,177,187,32,218,255,208,188,183
640 DATA208,248,32,204,255,78,208,168,32,59,192,168,72,32,201,885,189,87,32,218
650 DATA285,208,13,32,59,188,182,78,32,201,255,188,82,32,218,258,183,56,32,218
660 DATA255,32,212,225,188,8,177,187,32,218,255,208,188,188,288,240,188,13,32
670 DATA218,255,32,58,182,182,72,32,188,255,32,287,285,32,218,255,32,183,255
680 DATA201,84,208,243,32,58,168,78,248,188,159,8,133,18,32,212,225,188,8,133
690 DATA188,78,111,225,32,212,228,188,8,133,188,78,88,225,188,1,208,238,183,88
700 DATA134,88,132,144,56,32,73,188,32,223,125,133,253,132,254,188,255,188,8
710 DATA208,177,253,208,251,132,202,82,168,3,228,208,168,248,8,188,32,157,88
720 DATA3,232,136,202,246,177,253,157,58,3,232,208,881,8,208,245,238,158,32,187
730 DATA88,3,232,168,8,137,88,3,188,88,188,3,75,36,171,185,85,188,8,188,8,32
740 DATA185,255,165,1,188,88,188,163,32,188,205,32,182,255,188,88,32,188,255
750 DATA32,287,255,32,287,255,32,287,285,32,287,255,188,13,32,218,255,32,225
760 DATA255,240,184,32,183,255,208,88,32,287,255,133,854,32,287,255,182,254,32
770 DATA286,138,32,287,255,234,254,201,34,248,14,881,88,208,243,82,218,855,32
780 DATA287,255,888,208,240,188,168,8,157,125,2,232,32,887,285,187,125,2,201
790 DATA38,208,245,32,287,255,201,32,248,248,141,188,2,32,287,255,141,182,2,188
800 DATA32,181,184,2,234,188,6,137,185,2,162,255,232,188,182,2,248,5,32,218,285
810 DATA288,245,32,887,255,288,251,248,138,32,284,255,188,83,32,185,255,78,848
820 DATA188,125,4,133,125,188,147,141,118,2,168,82,141,128,8,188,213,141,121
830 DATA2,185,18,141,128,2,78,172,188,125,8,133,18,32,212,225,125,8,133,188,168
840 DATA1,133,185,78,111,225,41,207,133,178,36,178,88,3,24,185,8,41,15,88,32
850 DATA212,225,184,163,138,177,187,32,227,187,133,188,132,177,187,32,287,187
860 DATA18,18,18,18,24,181,188,133,188,138,177,187,32,827,187,132,188,132,177
870 DATA187,32,227,187,18,18,18,18,24,181,188,133,182,138,177,187,201,44,248
880 DATA5,188,11,78,138,827,132,177,187,32,227,187,133,881,132,177,187,32,827
890 DATA187,18,18,18,18,84,181,201,133,201,132,177,187,82,227,187,132,208,132
900 DATA177,187,32,227,187,18,18,18,18,24,181,208,133,252,138,177,187,881,44
910 DATA248,5,188,11,78,138,227,234,132,183,182,85,162,8,168,1,32,188,255,32
920 DATA182,255,188,85,32,881,255,185,251,32,218,253,125,208,82,218,253,188,8
930 DATA177,251,32,218,255,258,251,208,2,208,202,125,251,187,188,208,232,188
940 DATA252,187,125,208,233,32,204,203,125,85,32,155,235,78,148,188,32,212,228
950 DATA188,87,162,8,188,8,32,188,255,32,188,255,188,87,32,188,255,32,287,255
960 DATA133,232,32,287,255,133,253,188,232,234,32,208,188,188,13,32,218,255,238
970 DATA252,208,2,238,233,32,287,255,32,183,255,248,242,32,204,298,188,87,32
980 DATA185,285,168,208,185,263,32,285,188,168,13,32,218,208,78,148,188,32,255
990 DATA183,134,202,32,58,182,188,87,182,8,188,5,82,186,255,188,1,182,58,188
1000 DATA188,32,188,255,32,182,285,188,72,32,281,255,188,224,128,887,32,88,171
1010 DATA188,28,168,8,32,285,188,128,44,32,218,285,188,288,168,8,32,255,188,168
1020 DATA13,32,218,235,32,888,255,188,87,32,188,258,132,8,134,178,188,32,32,218
```

Listato Basic 3 - Questo è tutto l'ADP Basic Chi non ci ha seguito nei numeri scorsi, digiti direttamente questo listato

```
1030 DATA255 32 207,253 170 160,0 32 200 160 173,141,2 200,251,230,170,208,230
1040 DATA32 204 255 169,87 32 165 255,76,140 160 32,212 205,165 45 80,233,2,170
1050 DATA160 46 233 0 168,120,0 133 166,160,0,32 213 255 32,51,120 80,120,45
1060 DATA233 2,133,87,163,40 233,0 133,88,120 0 177 87,206,88,200,177,87 206
1070 DATA24 165 87 24 103 2 133,45 133,47 133,48,165,88 165 0,133,40,133,40 133
1080 DATA50 70,140,160 162 0,177 87 133,88 200 177,87,132 88,165,88,133 87 24
1090 DATA144 203 32 212,205 162 17 160 160 137 64 3 200,10,200 164 183,240 16
1100 DATA120,177 167 153 84 3 192 0 208 240,32 88,160 162,72,32 201,203,160,73
1110 DATA32 210 255 32 204 253 169 67,162 0,160 2,30,160,255,160 1,160,88 160
1120 DATA160 32 160 255,32,120 255 162 72,32 251,255,160,240,120 207,32,30 171
1130 DATA32 204 205 160 87,32 160 255 132 0 32,207,255 157,0,120 232,200 247
1140 DATA32 204,255 160 17 160,84,3,157,144 160 202,10,247,160,72,32,201 200
1150 DATA160,232,160,257 32 20 171 32,204 255,120,87 32,201,255,160,1,41,204
1160 DATA133,1,160,6,160,0 160,32,210 255 200,206 247 32,200 255 165,1,0,1 133
1170 DATA1 160,58,141,241 207 162,72,32,201,255,120,240 160 207 32,32 171 32
1180 DATA204,203,160,87,32 160,255,120,40 141,241,207 70,0,160 32 212,225,160
1190 DATA0 132 166 165,73 133 164,32 160 255,30 65,162 160,72,32,160,255,32,207
1200 DATA255 133,232,32 207 255 24 101,250 133,200,32 207 255,32,165,255 201
1210 DATA64,200,040 32 204,255,165 252,201 00,240 0,32 231,205,162 4,70,130,227
1220 DATA160,73 32 160,255,32 207 255 32 207 255,32,207,255,170,32 207,255,204
1230 DATA0,200 0 201,0 206,2,240 160,120 0,133,252,165 13,32,210,255 32,203,233
1240 DATA240,00,32,207,220,170 32,207,205,32 203,120,160,32,32,210,255 32 207
1250 DATA255,240,200,201 34,200 0,72,160,252 73 1,133 252,164 001,0,40,5 32,210
1260 DATA255,200 232,164 232,206,247,72,160,0 133,160 160,205 133,160,160 0 164
1270 DATA41,127,170 240 12 200 200 2,232 160,177 160 10 247 200,200 244 200,200
1280 DATA2,230,160 177,160,72 41 127 32 210,255,164,10,240 40,170,32,204 253
1290 DATA120,73,32 125,255 70,240,120,32,204 255,32,231,255,160,70,160,0 160
1300 DATA0,32,160,255,160 0,32,140,233,80,132 255,80 160,9,80,72 160,0,32 160
1310 DATA255,160 0 60,160 255,32,160,255,164,170,32,201,255,80,32 204,255 9,80
1320 DATA32 160 255 76,240 160,32,121 0 201,145,240 10,201,224,200 20,201,223
1330 DATA240 80,201,225 240,47 201 200 240 80 70 43,160 32 80 201,162,70,32,201
1340 DATA256,70,040,160,120,13,32,210 229,32,204,255,32,231 253,70,240,160,120
1350 DATA7,32,164,201 160,0,32,210,235,160,7 70,127,201 160 6,32,164,201 160
1360 DATA40,32,010 255,160,0,70,127,201 160,0,32,164 201,160 40 32 210 255 120
1370 DATA0,70 127,001 254 234,234 160 1,32,164,201,160 72,32,210,255 160 1,70
1380 DATA127,251 160,1 32 164,201 165 73 200 220 160 1,32,164,201 160 77,32,210
1390 DATA255 32,70,200,160 1,70,127,201,160,1,32,164,201,160,60 200 300 32,121
1400 DATA0,201 230,240,7,201,201,040,15 70,0,207,32,115,0 120,1 40 164 201,160
1410 DATA02,200 210,63,115,0,160,1,32,164 201 160,74 200 160 32 150,173 32,201
1420 DATA160 32 30 171 32 121,0,32 210,200,32 115,0 32,160,173 32,221,160,32
1430 DATA32,171 20 160,2,32,164 201 32 130 163 132,41 3,0 40,32,210,253,160,2
1440 DATA70,107 201 160 40,76 160,2,32 164,201,164 32 210 255 160,2,70,127,201
1450 DATA160,40,200,237,160 50,200 233,120,51,200 220 32,160,183 160,201,10 240
1460 DATA17 201,20,240 13,201,40 240,0 201,00 240 5 160,14,76,130,227,160 254
1470 DATA232 10,144,252,120 70,160 3,20,164 201 164 170,160,0,32 205,120,160
1480 DATA3 70,127,201 32 121,0,201 145,240 7,201 224,240,3,70,43 160,41,1,5,40
1490 DATA72,160 4 32,164,001 164 32 210 255 165,4,76,127 201,160 5,32,164,201
1500 DATA32,150,173,32,001,160,32,30,171,160,5,70,127,251,32,80,201,162,70,32
1510 DATA201,205,160,13,32,160 170 32,204 255 160,70 76,167,201,160,0 66,72,162
1520 DATA4,76,110,201,72,160 13 32,210,253 164,32,204 255 5,20,70,132 201,20
1530 DATA121,0,201,145,240 10 201,204,240 20,201,203 240,36,201,225,240 47,201
1540 DATA226,240 57,70,43,160 80 35 160 160 80 32,201 255,70,240 160,160 13,32
1550 DATA210,255,32,004,255,32,231 255,70,240,160,160 10 32,15,203,160,0,32,210
1560 DATA255 160,10,70,24,203,165 7,141,40 162,32,204,255,32 231,255,70 80,203
1570 DATA160 0,240 240,32,121,0,201 145 240 11,201,224 240,24 201,239,240 24
1580 DATA70,43 160 163 147,72 163 0,32,15 203,164 32,210,255 160 0,70,24,203
1590 DATA120 10,206,037,32,115,0,169,3 32 15 203 30,160,160 130,32 010 255,160
1600 DATA3,32,24,203,24,144,212,234,234,004,234,160,0,32,15,205,80,150,163,130
1610 DATA32,210,255,160,0,70,24,203,32,121,0,201 145,240,7,201 224,240,10,70
1620 DATA43 160 160,4 72 32,15,203,160,0,32,210 255 164,70 24,229,160,0 200 200
1630 DATA32,85,160,162,80,80,201,255,160,13 32,160,170 32,204,255,160,80 70,240
1640 DATA160,120,2,32 15,203,160 13,32,160,170,32,204 255,160,80 80,120,255,70
1650 DATA240 160 160,1,32,15 203,160,13,32,160,170,32,204,255,160,87,32 165,205
1660 DATA70,240,160,234,234 234 160,0,32,15 203,132,21,32,210,255,160,13,32,010
1670 DATA255,32,204,255 120 162,32,165,255,32,35,120,160 60,32 201,255,120,4
1680 DATA160 0,133,252 132,251,160 40,32,220,255,200,01,204 40,200,7,160 13,62
1690 DATA210,255,160 0,177,251,41,120,73,120,0,10,32,210,255,177,251,41,31,160
1700 DATA253,177 251,41,107,74,74,74,74 74,120,160,200 206,160 0 5 253 201 34
1710 DATA200,20,32,210,253,160,141 32 010,255,132,160,160 32,00,210,205,130,200
1720 DATA240,160,32,32,210 255,032,230,251,200 2,200 032 165,206,201,7,200,160
1730 DATA125,251 201 231,200 160 160,13,32,210,255,32,204 005,120 60,32,165,250
1740 DATA160,10,32,15,200 163 0 32,210,255,160,10,70 04,203,32,35 160 162,00
1750 DATA32,201,255,160,5,160,240,204,153,110,2,136,10,247,160 0,133 160,32 121
1760 DATA0,70,160 160,223,32,200 20 02 70 70,70,13,04,32,80,160,0,20,70,160,70
1770 DATA70,210 70,80 80,212,80 65 84,103,73,70,80,85,84 160,73,70,80,85 212
1780 DATA80 73,205,80 85 85,160 70,20,212,71,70,84,207 82,85,200,73,160,20,80
1790 DATA83,84,70,82,107,71,70 83 83,104,80 80,84 85,82,206,80 60,005 83,84,70
1800 DATA200,70,200,87,80,73,212,70 79,60,160,03,65 60,187,80,80,20,70,70 217
1810 DATA82 60 166 80,70,70,187,60,80 73,70,84,163,80,80,73,70,212,67,70 70,212
1820 DATA70,73,83 212,87,70 210 87,77 160,83,80,211,70,80,63,200,87 70,79,83
1830 DATA197,71 80,212 70,20,210,84 65,80,160,84,207,70,200,80,80,87,120,84,70
1840 DATA25 206,70,70,212,83,84,20 200 171,173,176,176,222,85,70,160 70,210,160
1850 DATA160,160,83,71,200 73 76,012,65,80,011,85,83,010,70,80,187,80,70,211
1860 DATA83,81,210,82 76,160,70,76,120 80 80 200 67,76,211 83,73,206,84,83 200
1870 DATA86,84 200 80 80,80 203,70,80 206,83,84 82,164,80,85,204,85,83 195,87
1880 DATA72 80,164,70,80,76 84,164 80,73,71,72,84,164 77,73,80,164,71,207,73
1890 DATA70,73,212,80 80,85 84,187 80,83 85 83 197,70 77,65,212,87,70,80,217
1900 DATA82,80,76,80,77,187,70,76,85,83 200,80,70,70 85,160,80 83,65,85,187,80
1910 DATA80,80,210,87,80,212,80 80,187,80,70,76,85,160 80,83,85,80,187,80 85
1920 DATA70,71,187 84,80,20,187,65,80 80,80,76,160,80,73 85,75,70,85 77,187,80
1930 DATA73,80,215,80,80,83 80,212,70,70,160,83,77,80,70 204,87,60 206,80,804
1940 DATA72,70,77,187,83 80,212,77,76,80,187,80,80,85,215,80,20,206,87,76,70
1950 DATA76,210 80 76,65,87 200,80,76,85 187,71,80 55,80 200,80,69,160,83,73
1960 DATA82,187,80,70,84,85,84,187 80 80,70,75 60 220,87 60 73,84,187,80,210
1970 DATA80,80,71,187,76,20 80,160,77,80,20,20,85,71,187 70,80,80,70,70,212,20
1980 DATA83,73,76,160,70,80,212,70,87,70,80,217,70,70,73,20,210,70,80,217,80
1990 DATA80 76,160 80 80 87,80 73,80 187,76 73,70,187,0,0,0,0,0,0,0,0,0,0,0,0
2000 DATA0,0,0,0,0 0,0,0,0,0 0,0,0,0 0,0,201,150 240,7,001,160,240,87,76,43,160
2010 DATA32,80,192 32,115 0,32,012,225,164 163 240,240,132,177,187 165,84,3,162
2020 DATA0 200,040,164,163,165,160 201,1,040 24 160,44 153,84,3,200,160 70,153
2030 DATA84,3,200 160,44,153 84,3,200,165,160 163 84,3,200 150,160 84,120,3,32
2040 DATA160 295 160 80,162 0,160,4,32 160,255,80,162,255,160,72,32,160,255,32
2050 DATA207 255,32,120,255 251 84 200,240,32,204,206 80,32 204,200,160 80,32
2060 DATA120,200 76,240,160 32 160,173,32,247,163,160,72,32,201,255,160,20,32
2070 DATA010,255,160 4,32 010 255,165,20,32 210,200,160,21,32 010,255,120,1,32
2080 DATA010,206,32 204,250,76,80,207 234,234,234,32,110,007,32,253,174,160,80
2090 DATA30,201,205,160,10 32,160,170,32,200,255 70,240,160,32,110,007 32,055
2100 DATA174,160 80,76,173,171,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
2110 DATA0,0,0,0 0,0,0,85,40,80,80,44,40 44,0,85,45,80,80,61,44,40,0,60,80,80
2120 DATA61,44,40,44,58,44,40,0,0,0,0,0
```

do INPUT, è obbligatorio che la RECEIVE sia inserita in una linea di programma, pena il messaggio d'errore ILLEGAL DIRECT.

Facciamo un esempio: scriviamo il seguente programmino per inserire alcune registrazioni in un file e per recuperarle agevolmente.

10 REL OPEN "ESEMPIO",50
con questa prima linea inizializziamo il file relativo, di nome ESEMPIO e con registrazioni di lunghezza massima 50 byte.

20 SEND 12,"BUONGIORNO"
riempiamo la 12-esima posizione con la stringa BUONGIORNO

30 SEND 3,"ARRIVEDERCI"
riempiamo la terza posizione con la stringa ARRIVEDERCI

40 REL CLOSE
chiudiamo il file precedentemente aperto.
Scriviamo ora un programmino per accedere alle registrazioni del file ESEMPIO e per modificarne alcune di esse:

10 REL OPEN "ESEMPIO"
questa volta non abbiamo indicato la lunghezza massima degli elementi, essendo il file già inizializzato

20 RECEIVE 12 A$
poniamo in A$ la 12-esima registrazione

30 PRINT A$
mostrerà su video la scritta BUONGIORNO, precedentemente prelevata dalla 12-esima posizione del file

40 SEND 5,"SALVE!"
riempiamo la quinta posizione con la stringa SALVE!

50 SEND 3,"ABBRACCI E BACI"
riempiamo la terza posizione con la stringa ABBRACCI E BACI, cancellando l'ARRIVEDERCI precedentemente inserito.

60 REL CLOSE
chiude il file aperto. È chiaro che non è possibile aprire più file relativi contemporaneamente, dato che le istruzioni SEND e RECEIVE non fanno riferimenti al nome del file in questione.

Il comando KEY attiva i tasti F1-F8 e mostra le definizioni correnti

Confessione e autocritica

Dobbiamo confessare una piccola cosa: l'ADP Basic, per come era stato pensato e ripensato al momento del concepimento, si sarebbe dovuto esaurire già da un mese. Trattamento file relativi e definizione dei tasti F1-F8 non erano in programma. Diciamo che la continuazione è avvenuta a gentile richiesta di alcuni lettori: "Un Tool che non definisce i tasti F1-F8?". "E i file

relativi che fine fanno?" "Per caso il 1541 di De Prisco non gestisce file relativi?" "Il comando VIEW non funziona coi file sequenziali" e molte altre frecciatine simili.

Va bene, eccovi accontentati. Anche se è stata dura lo spazio rimasto a disposizione nei 4K liberi del 64 era poco e malamente distribuito solo qualche buco qua e lá. Ció ci ha costretto a limitare la potenzialità di questi ultimi comandi, rimandando alla attenzione dell'utente il buon uso di questi. Ad esempio, il sistema non si accorge se un file relativo è stato aperto prima di usare SEND e RECEIVE e roba di questo taglio. Comunque, sempre e solo inconvenienti non distruttivi. Se qualcosa sembra non andare per il verso giusto, basta spegnere le periferiche, eseguire la sequenza Run/Stop e Restore, e riaccendere le periferiche.

Un'altra considerazione va fatta in merito ai comandi vuoti. Un comando vuoto é un comando che non c'e. In ADP Basic non sono ammessi comandi vuoti. Ci spieghiamo meglio se digitiamo

```
10 PRINT  CIAO  GOTO200
```

(notare i due duepunti di seguito) abbiamo inserito nella linea 10 un comando vuoto contenuto tra i due duepunti. Quando è attivo l'ADP Basic dando RUN sarà segnalato un SYNTAX ERROR IN 10. Allo stesso modo non si potrà scrivere su video

```
CAT
```

non essendo leciti i due punti iniziali (per la precisione abbiamo digitato un comando vuoto, seguito da duepunti e dal comando CAT). Il lettore si convincerà facilmente che perdere l'opportunità di inserire comandi vuoti non è nemmeno contemplabile.

Una limitazione un po' più seria, questa volta però é proprio colpa della Commodore, sta nella impossibilità di inserire comandi ADP di seguito al THEN di un IF. Ad esempio, la linea

```
10 IF A>0 THEN DRAW 100 100
```

non è lecita (a meno di un piccolo trucchetto) in quanto dopo il THEN, se non c'é un numero linea il controllo è passato alla normale interpretazione dei comandi Basic e non alla EXECUTE STATEMENT di una qualsiasi estensione Basic. Sarebbe bastato un salto indiretto in pagina tre, invece del salto diretto alla routine (Evviva! ho trovato un buco nel CMB BASIC! n.d.r.)

```
253 FORI=48356TO48378:READI1:POKEI,I1:NEXT
263 FORI=52305TO52345:READI1:POKEI,I1:NEXT
273 FORI=52470TO52470:READI1:POKEI,I1:NEXT
283 FORI=51903TO52454:READI1:POKEI,I1:NEXT:END
300 +++++++++++++++++++++++++++++++++++++++++++++++++++
310 +                                                 +
320 +   ***  ****  ****    ****  ***   ***  ***  ***  +
330 +  *   * *   * *   *   *   * *   * *    *   *    * +
340 +  ***** *   * ****    ****  ***** ***  *   *      +
350 +  *   * *   * *       *   * *   *     * *   *   * +
360 +  *   * ****  *       ****  *   * ***  ***  ***   +
370 +                                                 +
380 +              ------------------------------     +
390 +              (C)  1984      ADP SOFTWARE        +
400 +              ------------------------------     +
410 +                                                 +
500 +++++++++++++++++++++++++++++++++++++++++++++++++++
4003 DATA31,203,119,203,181,203,158,203,231,203,252,203,28,204,59,204,601,204
4013 DATA121,194,160,207,182,207
4503 DATA80,210,86,79,65,66,76,187,68,190,71,82,65,80,72,73,109,76,88,82,73,78
4513 DATA212,85,83,73,78,198,78,85,212,72,67,78,80,217,76,78,73,83,212
4533 DATA88,82,79,70,70,13,84,38,88,128
5003 DATA160,5,96,72,162,4,76,110,201,32,204,255,0,96,70,132,201,32,121,0,201
5013 DATA145,240,19,201,224,240,26,201,223,240,6,201,225,240,47,201,226,240,97
5023 DATA76,43,198,32,35,192,182,88,32,201,255,76,240,169,160,13,32,210,255,32
5033 DATA204,205,32,231,255,76,240,169,169,10,32,15,203,169,0,32,210,205,160
5043 DATA10,78,24,203,169,7,141,40,192,32,204,255,32,231,255,76,38,203,169,0
5053 DATA240,240,32,121,0,201,145,240,11,201,224,240,24,201,230,240,0,70,43,180
5063 DATA32,35,192,182,68,32,201,255,169,14,32,210,255,78,68,203,234,32,35,162
5073 DATA182,68,32,201,255,169,19,32,210,255,76,68,203,32,210,255,169,3,32,204
5083 DATA255,169,99,32,195,255,240,208,76,240,180,234,234,32,158,163,138,32,210
5093 DATA255,169,6,76,24,203,32,121,0,201,145,240,7,201,224,240,187,70,43,188
5103 DATA32,35,162,162,88,32,201,255,169,8,32,210,255,76,68,203,32,210,255,32
5113 DATA35,192,162,88,32,201,255,169,13,32,160,170,32,204,255,169,20,76,240
5123 DATA168,76,240,160,234,234,32,212,225,180,0,177,167,32,210,255,200,160,183
5133 DATA208,246,169,13,32,210,255,169,2,76,24,203,76,240,168,234,234,160,13
5143 DATA32,160,170,169,1,76,24,203,180,0,32,210,255,160,13,32,210,255,169,15
5153 DATA32,210,255,80,234,234,32,35,192,162,80,32,201,255,169,4,160,0,133,252
5163 DATA132,251,162,40,32,220,255,208,81,224,40,208,7,32,42,204,234,234,182
5173 DATA0,177,251,41,120,73,128,0,10,32,210,255,177,251,41,31,133,253,177,251
5183 DATA41,127,74,74,74,74,74,180,189,252,204,180,0,5,253,201,34,208,28,169
5193 DATA38,208,10,234,32,210,255,138,160,169,32,32,210,255,136,208,240,169,32
5203 DATA32,210,255,232,230,251,208,2,230,252,165,252,201,7,208,186,165,251,201
5213 DATA231,208,180,160,13,32,210,255,169,19,32,210,255,32,204,255,169,60,32
5223 DATA185,255,76,240,168,234,234,169,10,76,24,203,32,35,182,162,90,32,201
5233 DATA255,160,5,185,246,204,183,118,2,136,16,247,169,0,133,180,32,121,0,70
5243 DATA158,169
READY.
```

Listato Basic 4 - Chi non ci ha seguito nei numeri scorsi, e possiede una stampante MPS 801 oltre al listato Basic 3 deve mandare in esecuzione anche questo programma

Per ovviare a questo inconveniente basta inserire i ":" tra il THEN e il DRAW, digitando

```
10 IF A>0 THEN:DRAW 100,100
```

ma come? non avevamo detto che i comandi vuoti non erano leciti?

Esatto! Non dimentichiamo però che al momento del THEN il controllo passa al Basic standard che accetta comandi vuoti. La sequenza di operazioni è dunque questa: dopo il THEN non c'è un numero linea, il controllo è ceduto alla EXECUTE STATEMENT del Basic standard: trova i ":", non esegue nulla. Al giro successivo siamo nuovamente in ADP Basic, e possiamo eseguire il comando DRAW. Tutto procede secondo i piani.

Un'ultima cosa: se, usando l'ADP Basic, trovate qualche carenza o avete qualche suggerimento, scriveteci.

Dopo aver richiesto il Catalog, per caricare un programma basta posizionarsi col cursore e premere F5+RETURN

```
   ad ogni pressione del tasto F1, sara'
eseguito il list su video del programma
Basic contenuto in memoria.
   Per visualizzare il contenuto di un f
ile sequenziale, alla maniera del comand
o VIEW valido solo per i programmi basic
, e' disponibile il comando FILE, seguit
o dal nome del file.
   E' possibile anche leggere file dell'
EASY SCRIPT, precedentemente salvati dal
 word processor. Se non si desidera legg
ere tutto il file, e' sufficiente agire
sul tasto Run/Stop per interrompere ogni
tempo la visualizzazione.
   Per un trattamento facilitato dei fil
e relativi, sono disponibili 4 nuovi com
andi.
```

Col comando FILE è possibile leggere da disco un file sequenziale, ad esempio un file dell Easy Script

L'ADP Basic: riepilogo dei comandi

Comandi Disco

INIT - è usato per inizializzare il driver

VDATE - serve per convalidare i blocchi liberi di un dischetto

FMAT - "NomeDischetto, ID" formatta un dischetto: occorre specificare un nome e facoltativamente un identificatore di due caratteri. L'identificatore è obbligatorio solo se il dischetto è nuovo, ossia mai formattato

RENAME - "NuovoNome = VecchioNome" serve per cambiare nome a un file di un dischetto. NuovoNome e VecchioNome sono rispettivamente il nuovo nome e il vecchio nome del file preso in considerazione

ERASE - "NomeFile" Cancella un file da un dischetto. "NomeFile" è il nome del file da eliminare

COPY - "NuovoFile = VecchioFile" Copia sullo stesso dischetto il file "VecchioFile" dandogli come nome "NuovoFile"

COPY - "NuovoFile = VecchioFile1,VecchioFile2" Crea un nuovo file sul dischetto, a partire da 2 o più file già presenti, concatenandoli

DLOAD - "NomeProgramma" Permette di caricare programmi Basic da disco, senza la specifica "virgola otto"

DSAVE - "NomeProgramma" Salva su dischetto il programma Basic mantenuto in memoria

DVER - "NomeProgramma" Verifica la buona riuscita di un salvataggio su disco. Risponde con OK se tutto va bene o con VERIFY ERROR se è stato riscontrato un errore

FLASH - Mostra su video un eventuale messaggio del disco, segnalato dal lampeggio della spia rossa

CAT - Serve per visualizzare la directory di un dischetto su video, senza perdere il programma mantenuto in memoria

EXE - "NomeProgramma" "NomeProgramma" è il file da caricare e mandare in esecuzione. Equivale a un DLOAD seguito a fine caricamento da un RUN + [RETURN]

BLOAD - "NomeProgramma" Questo comando carica da disco un programma in linguaggio macchina, posizionandolo non all'inizio area Ram utente, ma nello stesso punto da dove era stato salvato. Corrisponde alla sequenza Basic LOAD "NomeProgramma",8,1

BSAVE - "NomeProgramma, Inizio, Fine" Serve per salvare programmi L.M. o più in generale sezioni di memoria, diverse da quella occupata da un programma Basic. L'indirizzo iniziale e finale vanno espressi in esadecimale, di 4 cifre, e (molto importante!) inseriti fra apici di seguito al NomeProgramma.

BANGE - "NomeProgramma" Serve per conoscere, senza caricare nulla, dove sarà posizionato un programma usando il comando BLOAD. Stampa su video in decimale l'indirizzo iniziale e finale

TRSE - Traccia, Settore. È usato per leggere un settore di una determinata traccia di un dischetto

APPEND - "NomeProgramma" Serve per saldare brutalmente due programmi Basic, uno in memoria e l'altro su disco. Brutalmente vuol dire che i numeri linea dei due programmi non hanno voce in capitolo: dopo l'ultima linea del primo programma è posizionata la prima linea del secondo

DISKNAME - "NuovoNome" Serve per cambiare il nome a un dischetto senza riformattarlo, ossia senza perdita di file

VIEW - "NomeProgramma" Si usa per visionare un programma, vedere il suo listato, senza caricarlo in memoria, direttamente dal disco su video. Per bloccare lo scroll è possibile agire sul tasto Run/Stop

REL OPEN - "NomeFile", LunghezzaElementi. Inizializza un file relativo di nome "NomeFile" e lunghezza degli elementi indicata dopo la virgola

REL OPEN - "NomeFile" Apre un file relativo precedentemente inizializzato

SEND - Posizione, "Stringa" Invia al file relativo nella posizione indicata la "Stringa"

RECEIVE - Posizione, VariabileStringa. Preleva dal file relativo la registrazione nella posizione specificata, ponendola nella variabile stringa

REL CLOSE - Chiude un file relativo, dopo la sua inizializzazione o dopo la sua lettura scrittura

FILE - "NomeFileSequenziale" Visualizza il contenuto di un file sequenziale presente sul dischetto

Comandi Plotter

PL - A seconda di cosa precede ha 5 diversi significati:

1) PL ON predispone il plotter come normale periferica di output. Corrisponde a un OPEN 4,6 seguito da CMD 4. Dopo tale comando qualsiasi PRINT sarà dirottata su periferica.

2) PL OFF riabilita il video quale periferica di output. Si usa per disattivare lo stato PL ON

3) PL RESET resetta il plotter, provocando il self test delle quattro penne colorate. Altri comandi inviati durante il reset vengono ignorati

4) PL SMALL predispone la scrittura con lettere minuscole normalmente e maiuscole se precedute da SHIFT. Se dato dopo PL ON, forza un PL OFF automaticamente (il video torna periferica di output)

5) PL CAP predispone la scrittura con lettere maiuscole normalmente e minuscole se precedute da SHIFT. Valgono le stesse considerazioni di PL SMALL. Usare PL ON solo dopo aver scelto le varie opzioni di stampa (colore penna, dimensione carattere, rotazione)

WRITE - Si usa per stampare direttamente una o più linee senza cambiare periferica di output. La sua sintassi è identica a quella del comando PRINT del Basic Standard del Commodore 64

SIZE - Serve per cambiare il formato dei caratteri. Si fa seguire da 10,20,40 o 80 a seconda del numero di caratteri per linea da stampare. Un parametro specificato diverso da quelli sopra elencati, genera il messaggio di errore ILLEGAL QUANTITY

ROTATE - Si usa per attivare la scrittura di caratteri orizzontali (per tabulazioni lungo il senso di scorrimento della carta). Più precisamente

ROTATE ON - ruota i caratteri di novanta gradi in senso orario

ROTATE OFF - seleziona il modo caratteri verticali (modo standard)

COLOR - NumeroColore. Cambia colore della penna del Plotter: seguito da 0 attiva la scrittura in nero, 1 attiva la scrittura in blu, 2 in verde, 3 in rosso

BLACK - predispone la scrittura in Nero

BLUE - predispone la scrittura in Blu

GREEN - predispone la scrittura in Verde

RED - predispone la scrittura in Rosso

MOVE - Ascissa, ordinata. Permette di spostare la testina di scrittura con la penna staccata dal foglio. Ascissa e Ordinata possono essere numeri, espressioni numeriche, variabili. Il plotter stesso ignora spostamenti oltre la cosiddetta Plottable Area, come ben spiegato sul manuale di istruzione dello stesso a pag. 16

DRAW - Ascissa, Ordinata muove la testina di scrittura mantenendo appoggiata sul foglio la penna. In altre parole traccia linee dalla attuale posizione sino alla posizione specificata

HOME - Serve per riportare la penna nel punto di coordinate (0,0) detto di origine

SET - Dichiara l'attuale posizione della testina di scrittura come nuovo punto di origine (0,0)

RELMOVE - Ascissa, Ordinata muove la testina di scrittura alle coordinate specificate secondo il nuovo sistema di riferimento dichiarato col comando SET

RELDRAW - Ascissa, Ordinata muove la testina di scrittura con la penna appoggiata sul foglio alle coordinate specificate, secondo il nuovo sistema di riferimento dichiarato col comando SET

BROKEN - NumeroTratteggio. Seguito da un numero compreso tra 0 e 15, permette di usare diversi tipi di tratteggio per le linee

Comandi Stampanti 801 e 802

PR - A seconda di cosa precede ha 5 diversi significati:

1) PR ON predispone la stampante come periferica di output. Dopo questo comando ogni normale PRINT sarà dirottata su carta

2) PR OFF disabilita lo stato di PR ON riattivando il video quale normale periferica di output

3) PR RESET invia alla Stampante 802 un comando di Reset. Serve per riportare in un sol colpo ai valori default tutti i parametri di stampa precedentemente variati

4) PR SMALL Predispone il set di stampa maiuscolo/minuscolo

5) PR CAP Predispone il set di stampa maiuscolo/grafico

Valgono le stesse considerazioni fatte per il comando PL del Plotter: PR SMALL o PR CAP deve essere selezionato prima del PR ON e mai dopo (un PR OFF è automaticamente forzato)

LPRINT - Permette un output diretto su stampante senza cambiare normale periferica di output. La sua sintassi è identica a quella del comando PRINT del Basic standard: dopo LPRINT può esserci una qualsiasi combinazione di stringhe e o numeri separati da virgola o punto e virgola

USING - "FormatoDiOutput" Comunica alla Stampante 802 il formato di output

OUT - "SequenzaDiDati" Invia alla Stampante 802 la sequenza di dati da stampare formattati. La sintassi è la stessa adoperata normalmente (senza l'ADP) ossia dialogando in termini di OPEN e PRINT#. Ricordiamo che CHR$(29) è il carattere di SKIP adoperato per effettuare il formattamento

PAGE ON - Attiva la paginazione automatica (salto di sei righe di stampa in prossimità della perforazione tra i fogli) per la Stampante 802

PAGE OFF - Disattiva la paginazione automatica

PAGE SIZE - NumeroRighe. Serve per variare il numero di righe per pagina da stampare prima del salto della perforazione tra fogli (solo 802)

FEED - Interlinea. Cambia l'interlinea (lo spazio tra le righe di stampa) della 802. Interlinea = 36 produce un avanzamento standard. Interlinea = 21, farà combaciare le matrici dei caratteri come avviene sul video del 64

MESSAGE ON - Attiva i messaggi di errore della Stampante 802

MESSAGE OFF - Disattiva la segnalazione di errori della Stampante 802

LLIST - Esegue il list su carta del programma Basic mantenuto in memoria

HCOPY - esegue la copia su carta del contenuto del video

DOUBLE ON - Attiva la stampa a caratteri espansi della 801

DOUBLE OFF - Torna al modo caratteri standard (801)

GRAPHIC ON - Passa al modo grafico della Stampante 801

GRAPHIC OFF - Torna al modo caratteri standard (801)

Comandi Vari

KEY - Attiva i tasti definibili dall'utente e visualizza le 8 stringhe associate ai tasti F1-F8

KEYn - "Stringa" Associa al tasto Fn la stringa indicata (possibilità di inserire anche CHR$(13) per forzare un comando)

OFF - disabilita l'ADP BASIC non rendendo più disponibili i nuovi comandi. Per rientrare in ambiente ADP è sufficiente digitare SYS 49152

RESET - provoca un vero e proprio RESET di tutta la macchina, reinizializzando tutte le variabili interne con relativa perdita del programma BASIC mantenuto in memoria. Anche dopo il RESET per riattivare l'ADP BASIC è sufficiente digitare SYS 49152.

Datazione automatica dei file su disco

Il fatto di leggere nella directory la data in cui il file è stato creato può sembrare una cosa inutile ma soprattutto per i programmi molto vecchi o per quei programmi di cui esistono più versioni avere la data scritta nel catalogo del dischetto fa risparmiare un mucchio di tempo

Per avere la data scritta sul disco occorre innanzitutto che il computer sappia la data odierna chi ha la scheda orologio calendario può far ricavare la data direttamente dal contenuto della variabile TI$, mentre chi non la possiede dovrà immettere la data da tastiera ad ogni boot del sistema. Una volta comunicata la data al computer, o più esattamente al programma Basic, questo si occupa di scriverla negli ultimi otto byte del Buffer di nome del DOS. Dopodiché la lunghezza utile dei nomi viene ridotta a ventun caratteri in modo che in fase di lettura la data non sia significativa. Così facendo il file PROVA con data 12 08 83 si carica semplicemente con LOAD PROVA senza dover riscrivere ogni volta la data. Viene poi apportata una piccola modifica al DOS per evitare che la routine di pulizia del Buffer di nome cancelli la data.

In pratica sostituendo il programma HELLO con questo, all'accensione del sistema viene chiesta la data odierna (se ne dite una vecchia comunque non si arrabbia) e, scritta questa nel DOS, viene lanciato in esecuzione il programma START che altri non sarebbe se non il vecchio HELLO opportunamente rinominato.

Ogni volta che si salva un file con un nuovo nome viene aggiunta la data in fondo al nome stesso. Come già detto la data non deve essere usata al momento del richiamo.

Avvertenze

Il programma di HELLO del disco *non* deve avere la data in quanto viene letto da un DOS non ancora "abituato" a questa novità.

Effettuando una RENAME il nuovo file sarà senza data. Si può mettere una data qualsiasi scrivendola direttamente nel secondo nome al posto giusto (ultimi otto byte) e nel giusto formato (gg mm aa).

Nel caso si voglia togliere una data, basta quindi scrivere

RENAME nome nome

Per cambiare data ad un file conviene eseguire

LOAD file
DELETE file
SAVE file

altrimenti il salvataggio di un file con un vecchio nome non modifica la data, a meno che il nuovo nome non sia lungo trenta caratteri.

```
DISK VOLUME 254

 A 004 HELLO
*A 004 START
*B 018 TEDAPPLE
*B 018 TEDITOR
*B 018 PRINTER
*B 019 PRT.HGR
*B 018 PRINTER_PAGE
 B 002 PRINT CONSTANTS
 B 002 PRINT CONST-HGR
 B 044 TEXT-RELENS        22.01.85
 B 007 TEXT.POSTA1        25.01.85
 B 006 TEXT.POSTA2        25.01.85
 B 019 TEXT.DATA          02.02.85
 B 008 TEXT.POSTA         19.12.84
 B 015 TEXT.RECE          19.12.84
 B 089 TEXT.PARK          10.01.85
 B 007 TEXT.ASS           20.01.85
```

Figura 1 - Esempio di Catalogo con file datati. Come potete notare il file HELLO non deve avere la data. Tutti gli altri possono invece avere la data o no senza che questo crei problemi al DOS. La data non va specificata mai tranne nel caso che si sia lanciato il DOS da un disco senza date

```
100  TEXT : HOME
110  INPUT "DATA ODIERNA (GG/MM/AA): ";DD$
120  IF  LEN (DD$)   8 THEN 530
130 GG =  VAL (DD$): IF GG < 1 OR GG > 31 THEN 530
135 G$ =  STR$ (GG): IF GG < 10 THEN G$ = "0" + G$
140  GOSUB 500
150 MM =  VAL (DD$)
155 M$ =  STR$ (MM): IF MM < 10 THEN M$ = "0" + M$
160  IF MM < 1 OR MM > 12 THEN 530
170  GOSUB 500
180 AA =  VAL (DD$)
190  IF AA < 0 OR AA > 99 THEN 530
195 A$ =  STR$ (AA): IF AA < 10 THEN A$ = "0" + A$
200 DD$ = G$ + "." + M$ + "." + A$
210  PRINT : PRINT "> "DD$" < ";
220  INPUT " O.K.? ";A$
230  IF  LEFT$ (A$,1) = "N" THEN  RUN
300  REM DOS.NAME= 21 CHR$
310  POKE 45571,20
320  REM  MODIFICA IL DOS
330  FOR I = 42367 TO 42384
340  READ A: POKE I,A
350  NEXT
360  POKE 41024,130: POKE 41025,165
370  FOR I = 1 TO 8
380  POKE 42385 + I, ASC ( MID$ (DD$,I,1)) + 128
390  NEXT
400  PRINT  CHR$ (4)"RUN START"
410 :
420  DATA  76,132,157,32,149,160,160,8,185,145,165,153
    ,138,170,136,208,247,96
430 :
500  FOR I = 1 TO  LEN (DD$)
510  IF  MID$ (DD$,I,1) = "/" THEN 540
520  NEXT : POP
530  PRINT  CHR$ (7);"INVALID FORMAT": GOTO 110
540 DD$ =  MID$ (DD$,I + 1)
550  RETURN
```

Listato 1 - Programma in Basic per la scrittura automatica della data in cui un file viene creato in fondo al nome stesso del file. Il suo uso non influenza minimamente le operazioni con i vecchi file che possono tranquillamente coesistere sullo stesso disco. Il programma deve chiamarsi HELLO mentre il vecchio HELLO deve prendere il nome di START

Note al programma Basic

La subroutine 500 controlla la validità del formato della data, da notare l'istruzione POP alla riga 520. Questa istruzione, poco nota e tipica dell'Applesoft, permette di uscire da una subroutine con un GOTO anziché un RETURN, infatti la sua funzione consiste nel togliere dallo Stack l'ultimo indirizzo di ritorno da subroutine. Senza la POP non è possibile uscire da un punto interno di una subroutine, e nel caso ci si attenda degli errori occorrerebbe settare un flag di errore (ad esempio una variabile ER) e controllare questo valore dopo ogni rientro da subroutine.

Le righe da 120 a 200 si occupano di convertire la data immessa in una lunga esattamente otto caratteri e controllano anche un minimo di coerenza (si potrebbe aggiungere il controllo del numero di giorni in funzione del mese).

Le righe da 300 a 390 preparano il DOS alla data e gliela scrivono dentro.

La riga 400 manda in esecuzione il vecchio HELLO che è stato rinominato START (RENAME HELLO,START).

La DATA della riga 420 contiene il nuovo programma di pulizia del Buffer del DOS, fate quindi attenzione a scrivere bene questi valori, come pure un po' di attenzione va prestata nei valori delle POKE. Un errore in questi numeri può distruggere il DOS in memoria e probabilmente anche il disco presente nel DRIVE, e buona abitudine usare un disco appena inizializzato per scrivere e provare tutte le routine che contengono delle POKE, anche se sono POKE che non coinvolgono il DOS!

```
0300- 86 FE 84 FB A5 B8 85 06
0308- A5 B9 85 07 20 E3 DF A0
0310- 02 B1 83 85 09 88 B1 83
0318- 85 08 88 B1 83 85 FC A8
0320- A9 20 99 2F 02 88 D0 FA
0328- A9 D1 85 FA A9 C8 85 FD
0330- A9 FB 85 F9 A2 00 84 FF
0338- E6 F9 D0 02 E6 FA A1 F9
0340- 29 7F D1 08 D0 11 A1 F9
0348- 30 03 C8 D0 EB A5 FD D0
0350- 16 E6 F9 D0 02 E6 FA A1
0358- F9 10 F6 A4 FF E6 FD A5
0360- FD C9 E7 90 D3 B1 08 99
0368- 30 02 C8 C4 FC 90 B9 A9
0370- 00 99 30 02 A6 FE A9 30
0378- 85 B8 A9 02 85 B9 20 67
0380- DD A5 06 85 B8 A5 07 85
0388- B9 A4 FB 4C 95 D9
```

Listato 1 - Codice oggetto della routine che permette l'interpretazione di una stringa come funzione

Conversione da Stringa a Funzione

Capita spesso nei programmi di matematica sia applicativi che educativi di sentire la mancanza della possibilità di immettere direttamente da tastiera la funzione matematica (anche complicata) che dovrà essere analizzata dal programma.

La maggior parte dei programmatori risolve il problema spiegando all'utente dove scrivere, nel programma, la sua propria funzione, generalmente si fa scrivere una riga tipo 200 DEF FN Y=...

A parte la macchinosità di un tal approccio si corre anche il rischio che vengano commessi degli errori, soprattutto da parte di utenti poco pratici, che possono portare, nel migliore dei casi, a blocchi del programma.

Un lettore, che ha già collaborato con la nostra rivista, ha risolto il problema in modo più brillante: ha spiegato al Basic come si può convertire una stringa in una funzione e calcolarne poi il valore.

```
                 1   BUFF      EQU $06
                 2   FRMNUM    EQU $DD67
                 3   PTRGET    EQU $DFE3
                 4   VARPNL    EQU $83
                 5   CHRGOT    EQU $B8
                 6   POINT     EQU $08
                 7   BUFFER    EQU $230
                 8   LEN       EQU $FC
                 9   YSAVE     EQU $FB
                10   TABL      EQU $F9
                11   TOKEN     EQU $FD
                12   XSAVE     EQU $FE
                13   YSAVE1    EQU $FF
                14             ORG $300
0300- 86 FE     15             STX XSAVE
0302- 84 FB     16             STY YSAVE
0304- A5 B8     17             LDA CHRGOT
0306- 85 06     18             STA BUFF
0308- A5 B9     19             LDA CHRGOT+1
030A- 85 07     20             STA BUFF+1
030C- 20 E3 DF  21             JSR PTRGET
030F- A0 02     22             LDY #2
0311- B1 83     23             LDA (VARPNL),Y
0313- 85 09     24             STA POINT+1
0315- 88        25             DEY
0316- B1 83     26             LDA (VARPNL),Y
0318- 85 08     27             STA POINT
031A- 88        28             DEY
031B- B1 83     29             LDA (VARPNL),Y
031D- 85 FC     30             STA LEN
031F- A8        31             TAY
0320- A9 20     32             LDA #$20
0322- 99 2F 02  33   NEXTC     STA BUFFER-1,Y
0325- 88        34             DEY
0326- D0 FA     35             BNE NEXTC
0328- A9 D1     36   NEXT      LDA #$D1
032A- 85 FA     37             STA TABL+1
032C- A9 C8     38             LDA #200
032E- 85 FD     39             STA TOKEN
0330- A9 FB     40             LDA #$FB
0332- 85 F9     41             STA TABL
0334- A2 00     42             LDX #$00
0336- 84 FF     43             STY YSAVE1
0338- E6 F9     44   NEXTCHAR  INC TABL
033A- D0 02     45             BNE NEXTC1
033C- E6 FA     46             INC TABL+1
033E- A1 F9     47   NEXTC1    LDA (TABL,X)
0340- 29 7F     48             AND #$7F
0342- D1 08     49             CMP (POINT),Y
0344- D0 11     50             BNE SCAN
0346- A1 F9     51             LDA (TABL,X)
0348- 30 03     52             BMI EXIT
034A- C8        53             INY
034B- D0 EB     54             BNE NEXTCHAR
034D- A5 FD     55   EXIT      LDA TOKEN
034F- D0 16     56             BNE SCRIVI
0351- E6 F9     57   SCAN1     INC TABL
0353- D0 02     58             BNE SCAN
0355- E6 FA     59             INC TABL+1
0357- A1 F9     60   SCAN      LDA (TABL,X)
0359- 10 F6     61             BPL SCAN1
035B- A4 FF     62             LDY YSAVE1
035D- E6 FD     63             INC TOKEN
035F- A5 FD     64             LDA TOKEN
0361- C9 E7     65             CMP #231
0363- 90 D3     66             BCC NEXTCHAR
0365- B1 08     67             LDA (POINT),Y
0367- 99 30 02  68   SCRIVI    STA BUFFER,Y
036A- C8        69             INY
036B- C4 FC     70             CPY LEN
036D- 90 B9     71             BCC NEXT
036F- A9 00     72             LDA #$00
0371- 99 30 02  73             STA BUFFER,Y
0374- A6 FE     74             LDX XSAVE
0376- A9 30     75             LDA # BUFFER
0378- 85 B8     76             STA CHRGOT
037A- A9 02     77             LDA # BUFFER
037C- 85 B9     78             STA CHRGOT+1
037E- 20 67 DD  79             JSR FRMNUM
0381- A5 06     80             LDA BUFF
0383- 85 B8     81             STA CHRGOT
0385- A5 07     82             LDA BUFF+1
0387- 85 B9     83             STA CHRGOT+1
0389- A4 FB     84             LDY YSAVE
038B- 4C 95 D9  85             JMP $D995
                86             LST OFF
```

Listato 2 - Sorgente Assembler della VAL-POTENZIATA

```
5  IF  PEEK (768) = 134 THEN 20
10  PRINT  CHR$ (4);"BLOAD SUPER-
     VAL.OBJO"
20  POKE 11,0: POKE 12,3
30  HOME : PRINT "DIMOSTRATIVO DI
     UNA ROUTINE IN L.M. CHE PER
    METTEIL CALOLO DI UNA ESPRES
    SIONE NU-MERICA CONTENUTA IN
     UNA STRINGA"
40  PRINT "INSERISCI L'ESPRESSION
    E NUMERICA CHE     VUOI RISOL
    VERE USANDO ANCHE FUNZIONI
      COMPLESSE (SQR, ,LOG,ETC..
    )"
45  PRINT : INPUT " >";A$
50  IF A$ = "" THEN 100
55 :
60 A =  USR (0)A$
65 :
70  PRINT : INVERSE : PRINT A$;"
     = ";A: NORMAL : GOTO 45
100  TEXT : END

 >SQR((SIN(2) 2)+(COS(2) 2))

SQR((SIN(2) 2)+(COS(2)^2)) = 1

 >INT(RND(1)*100+50)

INT(RND(I)*100+50) = 60
 >ABS(TAN(EXP(2*3.14)+LOG(SIN((SQR(9)-2) 3))/PDL(1)))

ABS(TAN(EXP(2*3.14)+LOG(SIN((SQR(9)-2)^3))/PDL(I))) = .290549713

 >I + A

I + A = 1.29054971
```

Listato 3. Programma dimostrativo dell'uso della VAL-POTENZIATA all'interno di un programma Basic. Ricordarsi di scrivere in testa al programma le POKE necessarie ad attivare la funzione USR(0)

Val-Potenziata

di Marco Merler - Gardolo (TN)

La routine di VAL-POTENZIATA permette di immettere una funzione qualsiasi (purche valutabile dall'Applesoft) direttamente con una INPUT F$ e restituisce il valore calcolato estattamente come se si fosse scritta una riga tipo F = <funzione>

La routine usa il comando USR (0) per passare la stringa alla VAL-POTENZIATA e per averne indietro il valore

La sintassi è

100 Y = USR(0)A$

naturalmente si possono usare per Y e A$ qualsiasi altra variabile

In testa al programma occorre definire i puntatori alla USR che corrispondono alle locazioni 11 e 12, nel caso nostro occorre eseguire una POKE 11,0 e una POKE 12,3 Se si sposta la routine in un'altra zona di memoria bisogna mettere in 11 e 12 il relativo valore (POKE12,IND/6 POKE11, IND-PEEK(12)*256)

Funzionamento della routine in L.M.

Una volta salvati i registri X ed Y e il puntatore al programma, la routine cerca in memoria la posizione della stringa che segue il comando USR(0) Una volta trovata, la strioga viene copiata in un Buffer (per comodita quello di tastiera) e vengono sostituiti i comandi Basic con i relativi Token A questo punto la "stringa" è pronta per essere valutata dalla routine $DD67, si rimettono a posto i puntatori del Basic e i registri della CPU e si rientra nell'Applesoft

Commento

Dal momento che la routine effettua il computo del contenuto di una stringa si puo usare, oltre che per gli input di funzioni, anche per calcolare il valore di espressioni direttamente create dal computer con le operazioni di estrazione e concatenamento di stringhe In pratica si puo calcolare il prodotto di due funzioni, una contenuta in A$ e l'altra in B$, semplicemente calcolando il valore di F$ = "(" + A$ + ") * (" + B$ + ")"

Pensate poi cosa si puo fare usando anche le varie MID$, LEFT$ e RIGHT$

LA NUOVA SERIE DEI MONITORI A COLORI DCD

DCD - La nuova serie dei monitori a colori che la Barco mette a Vostra disposizione per aiutarVi a risolvere problemi di collegamento sia alle tradizionali sorgenti di segnali video sia ai personal computer più diffusi.
Ogni modello dispone infatti di un numero di ingressi video tale da soddisfare ogni esigenza: video composito PAL, RGB Analogico, RGB TTL, ingresso audio, connettore J8 per videoregistratore.

La famiglia DCD
DCD 1640 (16") e DCD 2240 (22") per la Vostra workstation.

DCD 2740 (27") uno strumento audiovisivo indispensabile per l'addestramento, fiere, meeting.
La serie DCD è anche disponibile in versione Quadristandard.

TELAY INTERNATIONAL s.r.l.

COMPUTER GRAPHICS DIVISION
MILANO: Via L. da Vinci, 43 - 20090 Trezzano S/N
Tel. 02/4455741/2/3/4/5 - Tlx. TELINT I 312827
ROMA: Via Salaria, 1319 - 00138 Roma
Tel. 06/6917058-6919312 - Tlx. TINTRO I 614381

software VIC 20

Anche questo mese una rubrica ben fornita: ben quattro programmi.

Con il primo avrete a disposizione un buon numero di accordi per imparare a suonare la chitarra mentre con il secondo potrete dilettarvi a giocare a bowling. Il terzo è una trascrizione per il Vic dell'adventure "La piramide di Iumuh" e l'ultimo vi aiuterà a tenere in ordine il vostro archivio di programmi.

Accordi per chitarra

di Massimo Rodolfi - Bergamo

Mi chiamo Massimo Rodolfi e sono un appassionato lettore della vostra rivista. Vi scrivo per inviarvi un programma per il Vic 20 dotato di un'espansione di 8K (o 16K).

Non si tratta del solito videogame, ma di un utile ausilio per chi vuole imparare un buon numero di accordi per chitarra che, come molti sanno, sono tantissimi ed è abbastanza difficile tenerli a mente tutti, specialmente quando si è alle prime armi. Per questo motivo mi sono dedicato alla stesura di questo programma che consente una facile consultazione di ben 108 accordi.

L'uso è molto semplice. Dopo il consueto Run apparirà una schermata di presentazione che, se crea difficoltà, potrà essere tralasciata in fase di copiatura (linee da 12 a 30). Premendo un tasto qualsiasi si passa al menu principale che comprende le sette note, i giri armonici e l'End che consente di uscire dal programma. Un esempio: volendo ricercare l'accordo "La7" non si deve far altro che scegliere l'opzione "La" corrispondente alla pressione del tasto contrassegnato dal numero "6" e premere Return. Compiuta tale operazione, comparirà sullo schermo un secondo menu contenente la lista di tutti gli accordi di "La" disponibili e quindi si potrà selezionare il "La7" premendo il numero corrispondente, cioè il "13". Premendo ancora il Return verrà visualizzato l'accordo selezionato.

Voglio specificare che l'accordo e visualizzato così come lo vede chi imbraccia la chitarra e la piccola legenda specifica quali dita sono interessate e quali corde devono essere premute per ottenerlo. Per tornare al menu principale basterà premere un tasto qualunque; scegliendo l'opzione 8 di tale menu si otterrà una schermata contenente i primi sei giri armonici e per ottenere tutti gli altri basterà premere un tasto.

Commenti

Questo programma sarà certamente apprezzato da chi sta imparando la chitarra e nello stesso tempo si diletta con il computer. Gli accordi selezionati si presentano come quelli descritti sui manuali pratici, viene cioè visualizzata la prima parte del

```
4 REM ****************
5 REM *  ACCORDI PER *
6 REM *  CHITARRA    *
7 REM ****************
8 K$="" B$=""
9 C$="IIIIIIIII" D$=""
10 PRINT"" POKE36879 25 PRINT""
12 PRINT" ACCORDI PER CHITARRA "
13 PRINT"  BY RODOLFI MASSIMO"
14 PRINT""
15 PRINT""
16 PRINT""
17 PRINT""
18 PRINT""
19 PRINT"  I  IIIIIIIIIIIIII"
20 PRINT"  I  IIIIIIIIIIIIII"
21 PRINT""
22 PRINT""
23 PRINT""
24 PRINT""
25 PRINT""
26 PRINT PRINT PRINT"    PREMI UN TASTO    "
30 GETA$ IFA$=""THEN30
31 AA$="SCRIVI IL NUMERO       DESIDERATO"
32 BB$=" LISTA ACCORDI DI "
33 Y$="
34 Z$="
35 M$="DO  RE  MI  FA  SOL  LA  SI  GIRI ARMONICI  END"
40 PRINT""
50 PRINT"   LISTA DELLE NOTE   "
60 PRINT"" GOSUB11500
70 PRINT"" TAB(11)M$
80 PRINT"" AA$
140 INPUTZ
145 IFZ<1ORZ>9THEN140
150 ONZGOTO1000 2000 3000 4000 5000 6000 7000 8000 900
900 PRINT"CIAO"
910 PRINT"SPERO DI ESSERTI STATO UTILE" END
1000 PRINT"" BB$ "DO  " GOSUB11000
1010 PRINT"" FORI=1TO9 PRINTTAB(11)"DO" NEXT GOSUB1020 GOTO1120
1020 PRINT""TAB(13)Y$
1100 PRINT""AA$
1110 INPUTZ
1115 IFZ 1ORZ>9THENPRINT""   "" GOTO1110
1117 GOSUB10000
1118 RETURN
1120 ONZGOTO1200 1250 1300 1350 1400 1450 1500 1550 1600
1200 PRINTLEFT$(B$ 7)"DO+"
1210 PRINTRIGHT$(K$ 2) LEFT$(D$ 8)"I" RIGHT$(K$ 3)"" GOTO1990
1250 PRINTLEFT$(B$ 7)"DO-"
1260 PRINTRIGHT$(K$ 2) LEFT$(D$ 4)"1" GOTO1990
1300 PRINTLEFT$(B$ 7)"DO6"
1310 PRINTRIGHT$(K$ 4) LEFT$(D$ 8)"I" RIGHT$(K$ 3)"" GOTO1990
1350 PRINTLEFT$(B$ 7)"DO7"
1360 PRINTRIGHT$(K$ 4) LEFT$(D$ 8)"I" RIGHT$(K$ 3)"" GOTO
1990
1400 PRINTLEFT$(B$ 7)"DO7+"
1410 PRINTRIGHT$(K$ 4) LEFT$(D$ 7)"2" GOTO1990
1450 PRINTLEFT$(B$ 7)"DO7/5+"
1460 PRINTRIGHT$(K$ 4) LEFT$(D$ 11)"I" RIGHT$(K$ 3)"" GOT
O1990
1500 PRINTLEFT$(B$ 7)"DO-7"
1510 PRINTRIGHT$(K$ 2) LEFT$(D$ 9) B RIGHT$(B$ 3  4
1520 PRINT"" RIGHT$ D$ 5)"" RIGHT$(C$ 5) "
1550 PRINTLEFT$(B$ 7) DO5+
1560 PRINTRIGHT$(K$ 4) RIGHT$(D$ 9) 2
1600 PRINTLEFT$(B$ 7)"DO DIM"
1610 PRINTRIGHT$(K$ 2) RIGHT$(D$ 9)"4" GOTO1990
1990 GETZ$ IFZ$=""THEN1990
1999 GOTO40
2000 PRINT"" BB$ "RE  " GOSUB10500
2010 PRINT"" FORI=1TO10 PRINTTAB(11) RE" NEXT GOSUB2020 GOTO2210
2020 PRINT""TAB(13)X$ Y$
2190 PRINTAA$
2200 INPUT"" Z
2205 IFZ<1ORZ>18THENPRINT""   "" GOTO2200
2206 GOSUB10000
2207 RETURN
2210 IFZ>9THENZ=Z-9 GOTO2230
2220 ONZGOTO2300 2325 2350 2375 2400 2425 2450 2475 2500
2230 ONZGOTO2525 2550 2575 2600 2625 2650 2675 2700 2725
2300 PRINTLEFT$(B$ 7)"RE +"
2305 PRINTRIGHT$(K$ 2) RIGHT$(D$ 11)"B1" GOTO1990
2325 PRINTLEFT$(B$ 7) RE -"
2330 PRINTRIGHT$(K$ 4) RIGHT$(D$ 8) 3" GOTO1990
2350 PRINTLEFT$(B$ 7) RE 6"
2355 PRINTRIGHT$(K$ 2),RIGHT$(D$ 4)"4" RIGHT$(D$ 5 "I" GOTO19
90
2375 PRINTLEFT$(B$,7)"RE 7"
```

(continua a pagina 124)

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag 157

manico della chitarra rivoltato, in altre parole le corde indicate, dall'alto verso il basso, sono le seguenti Mi (cantino), Si, Sol, Re, La, Mi (basso) Le corde da premere nelle esatte posizioni sono evidenziate da un pallino nero con accanto un numero indicante il dito interessato ed il barre è indicato da una B, posta in alto sulla fila che va premuta Il programma si presenta molto bene, semplice da usare, elegante e controllando un buon numero degli accordi descritti non abbiamo riscontrato errori Per com'è scritto esso gira anche sul C64 sul quale bisogna però modificare l'output sullo schermo e la poke della linea 10 che cambia il colore del fondo

Bowling

di Giorgio Matelerkamp - Varese

Sono un ragazzo di 15 anni e vi invio un programma che penso possa interessarvi si tratta del gioco del bowling ed è scritto per un Vic 20 in configurazione base

Dato il Run, appare una schermata iniziale che chiede in quanti si vuol giocare e dopo aver digitato il numero di giocatori viene chiesto di inserire il nome di ognuno, a questo punto comincia il gioco vero e proprio

Sulla parte superiore dello schermo sono indicate le varie giocate, dieci in tutto, suddivise, come indicato nella riga immediatamente successiva, in due lanci ciascuna Se al primo tiro si fa centro e si buttano giù tutti i birilli (facendo così strike), si passa al tiro della giocata successiva (se si gioca da soli), oppure la "mano" passa ad un altro giocatore (se si gioca in più di uno) Giocando, comunque, tutto risulterà molto semplice

Sotto i numeri delle giocate verrà indicato il punteggio dei singoli lanci e per tirare si può utilizzare qualsiasi tasto Quando si arriva all'ultimo lancio viene concesso un bonus di un tiro se nella giocata precedente sono stati totalizzati dieci punti (cioè sono stati abbattuti tutti i birilli) oppure di due tiri se è stato realizzato uno strike

Commenti

Il programma proposto dal nostro amico rispecchia fedelmente le regole del gioco vero (la dinamica della caduta dei birilli è invece un po' diversa!) ed il punteggio viene calcolato in base ad esse Giocando in più di una persona compaiono delle piste in diversi colori, a seconda di chi sta tirando, i birilli sono visti in pianta in fondo alla pista, cioè vengono visualizzati come delle palline sistemate a triangolo Dall'altra parte vedremo un'altra pallina che spazia trasversalmente (cioè nel verso della larghezza) da una parte all'altra della pista e per lanciare questa pallina — che rappresenta la boccia — contro i birilli, bisogna premere un tasto al momento opportuno, cioè quando essa è il più possibile in prossimità del centro della pista vi assicuriamo che non è affatto semplice realizzare uno strike Quando si realizza uno spare (dieci punti con due tiri) o uno strike il punteggio viene aggiornato, com'è noto, dipendentemente dai punti totalizzati con i successivi tiri Il gioco è reso più piacevole dalla presenza del rumore che simula quello dei birilli colpiti dalla boccia

Piramide di Iunnuh

di Michele Morini - Monza (MI)

Sul programma non c'è molto da dire se non che esso è la traduzione fedele dell'adventure "Piramide di Iunnuh" realizzato da Aristide Torrelli e apparso su MC n 33 Tuttavia qualche riga ve la scrivo lo stesso, in quanto ho voluto correggere quelle che mi sono parse delle imperfezioni (o mia pignoleria?)

Le modalità di comunicazione e dialogo col computer sono rimaste immutate per cui rimando all'articolo citato per ulteriori chiarimenti

Questa traduzione, o trascrizione, gira su un Vic espanso con un minimo di 8K RAM e, contrariamente a quanto accade sul Texas, rimangono liberi circa 1600 byte per aggiunte di comandi, luoghi, schermate, suoni, ecc Il tutto è ovviamente trasportabile su C 64 senza modifiche, se escludiamo quella relativa alla Poke per settare il colore del video

Ma veniamo alle modifiche sono stati aggiunti due comandi, HELP e AIUTO che, contrariamente a quanto può sembrare, ottengono effetti diversi Il primo fornisce l'elenco delle abbreviazioni dei comandi impartibili al computer e AIUTO non ve lo dico

Se si vogliono aggiungere dei nuovi comandi non sorgono eccessivi problemi in quanto il programma è ben strutturato è sufficiente cambiare il valore della variabile NV (in testa al programma), aggiungere in coda ai data in linea 535 le abbreviazioni dei nuovi comandi e in riga 55 porre il numero di linea delle routine che li eseguono

Le altre modifiche riguardano la routine che esegue i comandi POSA, LASCIA, DAI· nella versione originaria essi non funzionavano per alcuni oggetti se non in determinati luoghi Adesso tutti gli oggetti ricevono democraticamente lo stesso trattamento Sono stati poi cambiati i nomi di alcune variabili poiché il Vic accetta identificatori di due caratteri significativi e la variabile ST è riservata al S O , altri cambiamenti sono dovuti alla traduzione degli if – then – else ed all'eliminazione dei data usati per definire le minuscole Infine una nota per le famigerate "abbreviazioni" degli statement CBM in questo programma ci sono linee che richiedono il loro uso, in particolare la linea 266 è zeppa come un uovo e abbreviata contiene 88 caratteri

Bowling

```
10 POKE36879 15 L 1 =7747 L 2 =7791 L 3 =7835 C$(1)="
11 C$(2)=C$(1)+"                C$ 3)=C$ 1)+"  C$=C$ 1
12 SC=36879 CO=30720 VI 36876 SC(1)=152 SC(2)=18 SC(3)=126
20 POKESC 27
30 PRINT " "C$ "     BOWLING"
40 PRINT"     ________________________"
110 PRINT "   G.M   SOFT  (C)1984"
111 PRINT"__________________________"
112 PRINT"     QUANTI GIOCATORI  1 3)"
113 GETA$ A=VAL A$ IFA<1ORA>3THEN113
118 PRINT"  SCRIVETE IL VOSTRO NOME  MAX   LARATTERI)"
120 FORX=1TOA
121 PRINT "GIOCAT.";X" "
122 INPUTR$(X)
123 A$ X =LEFT$(A$(X) 5) NEXT
130 PRINT"   1 2 3 4 5 6 7 8 9 10       FORY=1TO10 PRINT "     NEXT PRINT"  "
132 PRINT " ____________________ ____________________" ON-(A=1)GOTO138
134 PRINT"  ____________________  ON (A=2)GOTO138
136 PRINT" ____________________
138 PRINTLEFT$(C$ 11 +A$(1)
140 IFA>1THENPRINTTAB(10)+"    +A$(2)" "
150 IFA>2THENPRINTA$(3)"
154 PRINTLEFT$(C$ 11)+"  ______________________            ________________"
160 FORO=1TO10
165 FORZ9=1TOA POKESC SC(Z9)
167 FORX=1TO10 READV POKEV,81 POKEV+CO 6 NEXT RESTORE
168 J=0 G=0
170 GOSUB430 GOSUB550 P=L(Z9) GOSUB1000 L(Z9)=P
172 GOSUB1200
174 ON  L Z9 /2=INT(L(Z9)/2))GOTO169
175 IF(PEEK(L(Z9) 1)=47ANDO>10 =0THEN195
176 PRINTLEFT$(C$ 11)+"       "+A$(Z9)+"  FRI"
178 PRINT"ANCORA I TIRO"
179 FORX=1TO3000 NEXT
180 PRINTLEFT$(C$ 11)+ "       "+"
182 PRINT"
184 FOR =1TO10 READV POKEV 81 POKEV+CO 6 NEXT RESTORE
185 J=0 GOSUB430 GOSUB550 T(Z9)=T(Z9)+J PRINTC$(Z9) T(Z9)
186 K=J+48 IFJ+PEEK(L(Z9)-1)-224=10THENK=47
187 IFK=58THENK=152
190 POKEL Z9  K POKEL(Z9)+CO 4 GOTO225
194 U=0 IF(PEEK(L(Z9)-2)=152ANDO=10)=0THEN225
195 PRINTLEFT$(C$ 11)+"       "+A$(Z9)+"  FAI"
196 PRINT"ANCORA 2 TIRI"
198 FORY=1TO3000 NEXT
199 PRINTLEFT$(C$ 11)+ "       "+"            "
202 PRINT"
203 L(Z9)=L(Z9)-1
205 FORX=1TO10 READV POKEV,81 POKEV+CO,6 NEXT RESTORE
206 J=0 GOSUB430 GOSUB550 T(Z9)=T(Z9)+J PRINTC$(Z9) T(Z9)
207 K=J+176 IFK=186THENK=152
208 POKEL(Z9) K POKEL(Z9)+CO 4 L(Z9)=L(Z9)+1
210 IFPEEK(L(Z9)-1)=152THENFORX=1TO10 READV POKEV,81 POKEV+CO 6 NEXT RESTORE
211 U=U+1 ONUGOTO209,225
225 NEXTZ9 NEXTO
230 GOTO882
430 N=8123 I=22
440 POKEN 32 N=N+I IFN<7968ORN>8124THENI=-I
480 POKEN+30720 0-(PEEK(SC)=10) POKEN,81 GETA$ ON-(A$="")GOTO440 RETURN
550 N=N+1 GETA$
554 IFPEEK(N)<>81THEN595
556 Q2=N GOSUB610
560 POKEVI 150 POKEN-1 32 POKEN+CO 0 (PEEK(SC)=10) POKEN 81 POKEVI 0
565 FORW=1TO3
566 IFPEEK(N-21*W)=81THENQ2=N-21*W GOSUB610
568 IFPEEK(N+23*W)=81THENQ2=N+23*W GOSUB610
572 NEXT
595 N=N+1 POKEVI 150 POKEN-1 32 POKEN+CO.0-(PEEK(SC)=10) POKEN,81 POKEVI.0
597 IFN=18THEN612
599 FORSS=1TO50 NEXT GOTO550
610 J=J+1 POKEVI 210 POKEQ2 32 FORK=1TO50 NEXT POKEVI 0 FORK=1TO40 NEXT RETURN
612 N=0 POKEN 32 POKEN-1 32 RETURN
882 PRINTC$ "     FINAL SCORES "
883 PRINT "      A$ 1) T 1   IFA>1THENPRINT "      A$(2) T 2)
884 IFA=3THENPRINT "      A$(3) T 3)
887 PRINT"    ANCORA  S N)?
894 GETR$ IFR$=  THEN    894
895 IFR$="S"THENRUN
896 PRINT"  POKE36879,27 END
1000 O=J IFP/2<>INT(P/2)THENO=O+176
1002 IFO=186THENO=152
1004 IFP 2=INT(P 2)THENO=O+48
1006 IFO+PEEK P-1)-224=10THENO=47
1012 POKEP G POKEP+CO 4 IFO=152THENP=P+1 O=0
1100 P=P+1 RETURN
1200 REM
1201 T(Z9)=T(Z9)+J
1205 T(Z9)=T(Z9)-J*(PEEK(L(Z9)-2)=47)
1210 T(Z9)=T(Z9)-10* PEEK(L(Z9)-3)=47ANDPEEK(L(Z9) 2)=152)
1220 IFL(Z9)=7746ORL(Z9)=7790ORL(Z9)=7834THEN1290
1225 T(Z9)=T(Z9)-1*(PEEK L(Z9) 4)=152)
1227 T(Z9)=T(Z9)-J*(PEEK(L(Z9)-3)=152)
1228 T(Z9)=T(Z9)-J* (PEEK L(Z9)-5)=152ANDPEEK(L(Z9)-3)=152))
1230 T Z9)=T(Z9)-J*(PEEK(L Z9)-6 =152ANDPEEK(L(Z9)-4)=152ANDPEEK(L(Z9)-2)=152)
1290 PRINTC$(Z9) T(Z9)
1300 RETURN
2200 DATA8087,8020 8049 8051 8070,8072,8093 8095,8116 8139
```

Commenti

Dobbiamo cercare una piramide nel deserto, penetrarvi, trovare i tesori nascosti e portarli via! Naturalmente non staremo comodamente seduti davanti alla tastiera e al Tv ad essa collegato ed il computer sarà i nostri occhi e i nostri orecchi. Dovremo, in base alle informazioni fornitecì dalla macchina sul teleschermo, impartire delle adeguate istruzioni che ci aiutino ad esplorare l'ambiente circostante e a superare tutte le difficolta che si presentano. I luoghi accessibili sono 24 ed i vocaboli riconosciuti dal computer, tra verbi e sostantivi, sono circa settanta. I comandi vengono impartiti scrivendo un verbo seguito da un articolo o preposizione di due lettere e da un sostantivo, ad esempio "prendi la torcia". Alcuni comandi possono essere forniti in forma abbreviata (l'elenco è ottenibile con HELP) mentre altri possono essere impartiti in maniera diretta, cioè non nel contesto di una frase. Ad esempio "I" rende l'inventario degli oggetti raccolti durante il percorso, "ARRENDO" dichiara la resa del giocatore che non riesce a trovare la soluzione ed altri come "GUARDA" e "PUNTEGGIO". Le parole che la macchina è in grado di recepire sono le seguenti:

torcia, chiave, trave, scritta, porta, flauto, teschio, collana, scheletro, altare, benda, scure, batteria, droga, cobra, zaffiro, spada, vaso, sarcofago, maschera, mummia, sabbia, stanza, geroglifici, sala, corridoio, cunicolo, scala, nord, sud, est, ovest, n, s, e, o, su, giù, sopra, sotto, punteggio, arrendo, inventario, i, guarda, vai, corri, cammina, esamina, prendi, piglia, posa, lascia, dai, apri, *, suona, leggi, ondeggia, mangia, bevi, chiudi, rompi, sfonda, accendi, spegni, salta.

Il listato di questo programma per maggiore chiarezza viene riportato in caratteri maiuscoli e minuscoli per evitare i caratteri grafici poco comprensibili al posto delle maiuscole. Le linee da 2 a 7 producono una schermata di presentazione che, anche se di non difficile interpretazione, può anche essere omessa.

Per digitare il programma conviene porre il computer in modo "maiuscolo/minuscolo" premendo i tasti Shift e Commodore.

Lista programmi

di Dino Ticli - Lecco (CO)

Lista programmi è una utility per ordinare il proprio software ed ottenerne l'elenco su carta o su schermo. Ciò permette di tenere sempre sott'occhio i programmi posseduti insieme ad altre informazioni utili quali, ad esempio, una breve descrizione sul funzionamento, l'espansione necessaria, il linguaggio usato, la collocazione, ecc. Vediamo rapidamente il funzionamento

3 *(segue da pagina 127)*

```
[illegible]
```

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 157

```
102 PRINT "J INSERISCI  IL NASTRO    NEL REGISTRATORE
103 PRINT    E PREMI  UN TASTO
104 GET X$ IFX$= "THEN104
105 PRINT"   CHE FILE CERCO   INPUTFI$
106 OPEN1 1 0 FI$ J=1
107 INPUT#J I
108 FORT=1TOI INPUT#J N$(T) INPUT#J,NO$(T) INPUT#J,DE$(I) INPUT#J EX$(T) GOSUB1
8 NEXT
109 CLOSEJ --I+1 GOTO17
110 PRINT "J   REGISTRAZIONE DATI
111 PRINT   SEI SICURO ?"
112 GET X$ IFX$=""THEN112
113 IFX$<>"S THEN32
114 INPUT"   NOME FILE",FI$
115 OPEN1 1 1 FI$ J=1
116 PRINT#J I FORT=1TOI PRINT#J N$(T) PRINT#J,NO$(T) PRINT#J,DE$(T) PRINT#J EX$(
T)
117 GOSUB118 NEXT CLOSEJ GOTO9
118 PRINT            SPC(17)"@" T RETURN
```

Linee sostitutive per usare il registratore nel programma "Lista programmi"

Innanzitutto è necessaria un'espansione di almeno 16K Ram per poter gestire circa 200 programmi. Dopo il Run verranno sottoposte due scelte: iniziare una nuova lista o leggerne una vecchia per aggiornarla o stamparla. In fase di aggiornamento o di descrizione verrà richiesto, dopo la comparsa del numero progressivo, il nome, massimo 22 caratteri, la descrizione, massimo 70 caratteri, i codici, massimo 12 caratteri; in quest'ultima voce io ho inserito i dati relativi all'espansione di memoria necessaria, al linguaggio usato, alla collocazione. Ad esempio: 16K-BA-D001, va interpretato come 16K di espansione, programma in Basic, disco n. 001.

Nella descrizione si possono usare i due punti e la virgola, pur essendo in fase di Input, grazie a un piccolo trucco facilmente comprensibile osservando la linea 29: i due Poke servono per aprire le virgolette cosicché virgola e due punti che eventualmente seguissero non sono più considerati segni di punteggiatura utili al Basic, ma elementi della stringa. Non è necessario chiudere le virgolette alla fine.

Sempre in questa fase si possono notare, nella parte alta dello schermo, delle opzioni: col tasto "↑" si passerà alla registrazione dei dati, con "@" al menu principale e con "←" all'ordinamento alfabetico degli stessi.

Il menu principale da la possibilità di rivedere le voci inserite ed eventualmente di correggerle molto agevolmente, inoltre si può passare in fase di stampa, registrazione o tornare a quella di creazione lista.

Come detto, è possibile inserire circa 200 record, ma questa cifra dipende solo dalla memoria utilizzata per ciascun dato. Meno ci si avvicina ai limiti indicati riguardo al numero di caratteri e più dati è possibile inserire.

Tengo a precisare che il programma descritto gira perfettamente anche sugli altri modelli Commodore e cioè C64, PLUS4 (su cui si potrà aumentare la dimensione delle matrici) e C16 (su cui la diminuiremo). Vanno naturalmente modificate anche le linee 17 e 44.

Per finire, il programma prevede l'uso del disk driver, ma sostituendo le linee da 102 a 118 con quelle del listato presentato a parte, è possibile utilizzare il registratore.

Commenti

Il programma è molto ben strutturato e non ha bisogno di ulteriori commenti. Si può passare da una sezione all'altra molto agevolmente ed altrettanto agevole è effettuare delle correzioni, inoltre i messaggi che compaiono sullo schermo di volta in volta non lasciano dubbi sul modo di operare.

Quando si procede alla registrazione del file su disco, viene conservato il numero dell'ultimo record inserito per cui, dopo un'eventuale lettura, il puntatore del file relativo verrà posizionato automaticamente in coda all'ultimo record inserito.

Finestra grafica

di Alberto Ghizzoni - Segrate (MI)

L'idea di partenza per la realizzazione di questo programma mi è nata dagli accenni fatti dalla Programmer's Guide del 64 in particolare, un paragrafo accennava all'interrupt del raster (che controlla la generazione dell'immagine video da inviare al televisore) e anche ai modi grafici misti. Il programma da me realizzato serve a dividere lo schermo orizzontalmente, creando una finestra indipendente (in pratica un secondo schermo) dalle seguenti caratteristiche

- ordinata variabile,
- regolazione fine dell'ascissa,
- posizione sullo schermo variabile,
- gestione sia da tastiera che da programma (in LM, ovvero con le POKE),
- estrema facilità di modifica,

va inoltre specificato che il tutto non interferisce in alcun modo con le normali funzioni del computer.

Come ho già detto, alla base di tutto sta l'interrupt del raster, che periodicamente confronta il suo contatore con il contenuto del registro che sfrutta le locazioni 53266-53267 quando i valori sono uguali viene generato un segnale di interrupt, segnale che può comunque essere mascherato. A questo punto una routine esamina il registro dell'interrupt del raster e dirama di conseguenza. Se la causa è effettivamente il raster si esegue la solita routine, altrimenti esegue un CLI (per ridare le giuste priorità di servizio) e passa il controllo all'usuale routine di gestione degli interrupt, posta in $EA31. Il nostro compito è quindi, grosso modo, il seguente: inserirci nella routine di generazione dello schermo, fargli generare una prima parte (che prende i dati da una zona di memoria A), verificare quando finisce la finestra, sostituire la zona A con un'altra B, fargli terminare il quadro televisivo, rifargli vedere la zona A e così via. In questo modo possiamo gestire due pagine completamente indipendenti, ad esem-

> Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 157

Listato 1

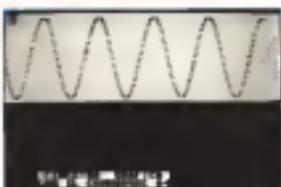
```
10 REM********************************
20 REM                                *
30 REM       FINESTRA  GRAFICA        *
40 REM                                *
50 REM           PER C-64             *
60 REM                                *
70 REM      DI ALBERTD GHIZZONI       *
80 REM                                *
90 REM********************************
91 REM
95 REM
100 FDRT=49152TO49209:GDSUB1000:PDKET,C:
NEXT
110 DATAA5,FC,8D,12,D0,AD,11,D0
120 DATA29,7F,8D,11,D0,A9,46,8D
130 DATA89,C1,8D,89,C3,A9,3F,85
140 DATAFO,85,FE,A9,C1,8D.15,03
150 DATAA9,83,8D,14,03,A9,00,85
160 DATA02,A9,8F,8D,19,D0,20,3A
170 DATAC0,AD,1A,D0,09,01,8D,1A
180 DATAD0,60
190 REM--------------------------------
200 FDRT=49587TO49599:GDSUB1000.PDKET,C:
PDKET+512,C:NEXT
210 DATAAD,19,D0,29,01,D0,46,AD
220 DATA0D,DC,58,4C,31
230 REM--------------------------------
240 FORT=49600TO49663:POKET,234.POKET+51
2,234:NEXT
250 REM--------------------------------
260 FORT=50432TD50598.GOSU81000:PDKET,C:
NEXT
270 DATAE6,02,A5,02,29,01,F0,16
280 DATAA5,FC,8D.12,D0,A5,FE,29
290 DATA3F,18,69,07,8D,89,C3,A9
300 DATAC3,8D,15,03,D0,17,A5,F8
310 DATA8D,12,D0,A5,FD,29,3F,18
320 DATA69,07,8D,B9,C1,A9,C1,8D
330 DATA15,03,20,40,C5,A9,8F,8D
340 DATA19,D0,68,A8,68,AA,68,40
350 DATAA5,C5,C9,04,F0,0D,C9,05
360 DATAF0,15,C9,06,F0,1D,C9,03
370 DATAF0.23.60,AD,8E,02,29.04
380 DATAD0,25,C6,F8,C6,FC,60.AD
390 DATA8E,02,29,04,D0,23,E6,F8
400 DATAE6,FC,60,AD,8E,02,29,04
410 DATAD0,21,C6,FC,60,AD,8E,02
420 DATA29,04,D0,21,E6,FC,60,A4
430 DATAFD,C8,C0,40,F0,02,84,FD
440 DATA60,A4,FD,88,C0,FF,F0,02
450 DATA84,FD,60,A4,FE,C8,C0,40
460 DATAF0,02,84,FE,60,A4,FE,88
470 DATAC0,FF,F0,02,84,FE,60
480 REM--------------------------------
490 REM--------------------------------
500 PDKE49210,96
510 REM--------------------------------
520 FDRT=49664TD49679.GDSU81000-POKET,Ç.
NEXT
530 DATAA9,0D,8D,21,D0,AD.18,D0
540 DATA09,02,8D,18,D0,4C,00.C5
550 REM--------------------------------
560 FORT=50176TD50191:GDSU81000:POKET,C:
NEXT
570 DATAA9,06,8D,21,D0.AD,18,D0
580 DATA29,F0,8D,18,D0,4C,00,C5
990 END
1000 READA$:A=ASC(LEFT$(A$,1))-48.1FA>9T
HENA=A-7
1010 8=ASC(RIGHT$(A$.1))-48:1F8>9THEN8=8
-7
1030 C=A*16+B.RETURN
```

Questa versione, la n. 1, usa la finestra per avere un diverso colore di sfondo e il set dei caratteri minuscoli

pio una grafica e una testo, ma anche una di un colore (e con un suo set di caratteri) e una di diversa tonalità. L'unico problema era la gestione dell'alternanza di modi: ho risolto la questione facendo sì che ognuna delle due routine da eseguire in rapida successione predisponga il vettore IRQ per l'interrupt successivo, svolgendo così le operazioni.

A grandi linee il programma è diviso in 7 blocchi:

(1) preparazione C000-C039,
(2) routine dell'utente C03A-C102
(3) diramazione 1 C103-C1FF,
(4) prima routine grafica C200-C302,
(5) diramazione 2 C303-C3FF,
(6) seconda rout. grafica C400-CFF,
(7) controllo tastiera C500-C559,

le due routine di diramazione devono terminare con un JMP C500, mentre quella dell'utente — che nel programma pulisce la pagina grafica, ma può essere agevolmente

sostituita a piacere — deve terminare con un RTS. Le (1) e (2) vengono eseguite una sola volta, mentre la (5) una volta sì e una no, altrimenti risulta troppo veloce.

Vediamo adesso come controllare le dimensioni della finestra, che si basa sui contenuti di 4 locazioni libere in pagina zero, e precisamente quelle da 251 a 254. Nella prima c'è la coordinata verticale dell'inizio, e nella successiva 252 ho messo quella finale; in 253 c'è la coordinata orizzontale della linea iniziale della finestra, e in 254 c'è quella finale. Per queste ultime due locazioni la questione è più complessa di quanto sembri, a causa della instabilità del punto di attacco, che rende necessario l'inserimento nel programma di un ritardo calcolato che tenga conto della variabilità di questo tempo: a tale scopo, tra la (3) e la (4), ma anche tra la (5) e la (6), si trova una pista di 63 NOP, dei quali il programma esegue solo quelli che servono (il numero esatto viene calcolato in modo semplice), ovvero — in pratica — il programma modifica se stesso. Per modificare questi valori da programma tenete appunto conto del fatto che il contenuto delle locazioni 253 e 254 non può eccedere 63; per maggior sicurezza ho comunque messo un AND $63. In 251 e 252, invece, potete mettere valori qualunque. Comunque sia, il programma

Listato 2

```
10 REM*********************************
20 REM                                *
30 REM      FINESTRA  GRAFICA         *
40 REM                                *
50 REM            PER C-64            *
60 REM                                *
70 REM     DI AL0ERTD GHIZZDNI        *
80 REM                                *
90 REM*********************************
91 REM
95 REM
100 FDRT=49152TD49209:GDSU01000:PDKET,C:
NEXT
110 DATAA5,FC,0D,12,D0,AD,11,D0
120 DATA29,7F,0D,11,D0,A9,46,0D
130 DATA09,C1,0D,09,C3,A9,3F,05
140 DATAFD,05,FE,A9,C1,0D,15,03
150 DATAA9,03,0D,14,03,A9,00,05
160 DATA02,A9,0F,0D,19,D0,20,3A
170 DATAC0,AD.1A,D0,09,01,8D,1A
100 DATAD0,60
190 REM-------------------------------
200 FDRT=49507TD49599:GDSU81000:PDKET,C:
PDKET+512,C:NEXT
210 DATAAD,19,D0,29,01,D0,46,AD
220 DATA0D,DC,50,4C,31
230 REM------------------------------
240 FDRT=49600TD49663:PDKET,234:PDKET+51
2,234:NEXT
250 REM------------------------------
260 FDRT=50432TD50590:GDSU01000·PDKET,C:
NEXT
270 DATAE6,02,A5,02,29,01,F0,16
200 DATAA5,FC,0D,12,D0,A5,FE,29
290 DATA3F,10,69,07,0D,09,C3,A9
300 DATAC3,0D,15,03,D0,17,A5,F0
310 DATA0D,12,D0,A5,FD,29,3F,10
320 DATA69,07,0D,09,C1,A9,C1,8D
330 DATA15,03,20,40,C5,A9,0F,0D
340 DATA19,D0,68,A0,68,AA,60,40
350 DATAA5,C5,C9,04,F0,0D,C9,05
360 DATAF0,15,C9,06,F0,1D,C9,03
370 DATAF0,23,60,AD,0E,02,29,04
300 DATAD0,25,C6,F8,C6,FC,60,AD
390 DATA0E,02,29,04,D0,23,E6,F0
400 DATAE6,FC,60,AD,0E,02,29,04
410 DATAD0,21,C6,FC,60,AD,0E,02
420 DATA29,04,D0,21,E6,FC,60,A4
430 DATAFD,C0,C0,40,F0,02,84,FD
440 DATA60,A4,FD,00,C0,FF,F0,02
450 DATA04,FD,60,A4,FE,C8,C0,40
460 DATAF0,02,04,FE,50,A4,FE,80
470 DATAC0,FF,F0,02,04,FE,60
400 REM-------------------------------
490 REM-------------------------------
500 FDRT=49210TD49261:GDSU01000:PDKET,C:
NEXT
510 DATAA9,5C,05,FE,A2,04,A0,00
520 DATA04,FD,A9,0D,91,FD,C8,D0
530 DATAF0,E6,FE,CA,D0,F6,A2,20
540 DATA98,91,FD,C0,D0,F8,E6,FE
550 DATACA,D0,F6,05,33,05,37,A9
560 DATA5C,05,34,85,30,A9,3F,85
570 DATAFD,05,FE,60
580 REM-------------------------------
590 FDRT=49664TD49702:GDSU81000:PDKET,C:
NEXT
600 DATAAD,02,DD,09,03,0D,02,DD
610 DATAAD,00,DD,29,FC,09,02,0D
620 DATA00,DD,AD,18,D0,09,70,8D
630 DATA18,D0,AD.11,D0.29,7F,09
640 DATA20,0D,11,D0.4C,00,C5
650 REM-------------------------------
660 FDRT=50176TD50212:GDSU01000:PDKET,C:
NEXT
670 DATAAD,02,DD,09,03,8D,02,DD
600 DATAAD.00,DD.29,FC,09,03,8D
690 DATA00,DD.AD,10,D0.29,15,0D
700 DATA10,D0,AD,11,D0,29,5F,0D
710 DATA11,D0.4C,00.C5
990 END
1000 READA$:A=ASC(LEFT$(A$,1))-40:IFA>9T
HENA=A-7
1010 0=ASC(RIGHT$(A$,1))-48:IF0>9THEN0=0
-7
1030 C=A*16+0:RETURN
```

La vesione n. 2 abilita una pagina in alta risoluzione, nella quale si può scrivere agevolmente da Basic (vedere esempio)

Listato 3

```
3000 REM ** STAMPA I  NUMERI DI RIGA
3010 PRINT"□" FORT=2TO20 PRINTT NEXT
4000 REM ** VISUALIZZA UNA SINUSOIDE
4001 REM ** NELLA  FINESTRA  GRAFICA
4040 BASE=24576
4050 FOR X=0 TO 300
4060 V=INT (45+40*SIN(X/10))
4070 CH=INT(X/8) RO=INT(V/8) LN=VAND7
4080 BY=BASE+RO*320+8*CH+LN
4090 BI=7-(XAND7)
4110 POKE BY,PEEK(BY)OR(2↑BI)
4120 NEXT
4125 POKE 1024,16
4130 PRINT "□□□□□□□□□□□□□□□□□□□□□□□□"
4140 PRINT"      ■ FINESTRA  GRAFICA ■"
4150 PRINT"      ■CON IL COMMODORE 64■"
4999 GOTO 4999
```

In questo esempio la mezza pagina superiore mostra una sinusoide mentre la metà inferiore è adibita a testo

prevede anche l'uso dei tasti funzione (e del CTRL).

F1 la sposta verso l'alto,
F2 la sposta verso il basso,
F5 diminuisce l'ampiezza,
F7 aumenta l'ampiezza

Gli stessi tasti, premuti contemporaneamente al CTRL, servono a

F1 spostare a sinistra il punto orizzontale,
F3 spostarlo a destra,
F5 a far arretrare il punto finale,
F7 a farlo avanzare

I valori in 251, 252, 253 e 254 non vengono modificati dalla routine iniziale, quindi se non vengono cambiati dopo l'accensione, contengono tutti 0. Se la finestra è troppo piccola (meno di 4 linee in alta risoluzione) o troppo grande (più di 251) gli interrupt sono troppo ravvicinati e le relative routine impediscono, per la loro dura-

Disassemblato 1

```
.C000 A5 FC       LDA $FC
.C002 8D 12 D0    STA $D012
.C005 AD 11 D0    LDA $D011
.C008 29 7F       AND #$7F
.C00A 8D 11 D0    STA $D011
.C00D A9 46       LDA #$46
.C00F 8D 89 C1    STA $C189
.C012 8D B9 C3    STA $C3B9
.C015 A9 3F       LDA #$3F
.C017 85 FD       STA $FD
.C019 85 FE       STA $FE
.C01B A9 C1       LDA #$C1
.C01D 8D 15 03    STA $0315
.C020 A9 B3       LDA #$B3
.C022 8D 14 03    STA $0314
.C025 A9 00       LDA #$00
.C027 85 02       STA $02
.C029 A9 8F       LDA #$8F
.C02B 8D 19 D0    STA $D019
.C02E 20 3A C0    JSR $C03A
.C031 AD 1A D0    LDA $D01A
.C034 09 01       ORA #$01
.C030 8D 1A D0    STA $D01A
.C039 60          RTS


.C183 AD 19 D0    LDA $D019
.C186 29 01       AND #$01
.C180 D0 46       BNE $C230
.C1BA AD 0D DC    LDA $DC0D
.C1BD 58          CLI
.C1BE 4C 31 EA    JMP $EA31

C1C1 - C1FF : 63 NOP

.C383 AD 19 D0    LDA $D019
.C386 29 01       AND #$01
.C380 D0 46       BNE $C400
.C38A AD 0D DC    LDA $DC0D
.C38D 58          CLI
.C38E 4C 31 EA    JMP $EA31

C3C1 - C3FF : 03 NOP

.C500 E0 02       INC $02
.C502 A5 02       LDA $02
.C504 29 01       AND #$01
.C506 F0 16       BEQ $C51E
.C508 A5 FC       LDA $FC
.C50A 8D 12 D0    STA $D012
.C50D A5 FE       LDA $FE
.C50F 29 3F       AND #$3F
.C511 18          CLC
.C512 69 07       ADC #$07
.C514 8D B9 C3    STA $C3B9
.C517 A9 C3       LDA #$C3
.C519 8D 15 03    STA $0315
.C51C D0 17       BNE $C535
.C51E A5 FB       LDA $FB
.C520 8D 12 D0    STA $D012
.C523 A5 FD       LDA $FD
.C525 29 3F       AND #$3F
.C527 18          CLC
.C528 69 07       ADC #$07
.C52A 8D B9 C1    STA $C1B9
.C52D A9 C1       LDA #$C1
.C52F 8D 15 03    STA $0315
.C532 20 40 C5    JSR $C540
.C535 A9 0F       LDA #$0F
.C537 8D 19 D0    STA $D019
.C53A 68          PLA
.C53B A8          TAY
.C53C 68          PLA
.C53D AA          TAX
.C53E 68          PLA
.C53F 40          RTI
.C540 A5 C5       LDA $C5
.C542 C9 04       CMP #$04
.C544 F0 0D       BEQ $C553
.C546 C9 05       CMP #$05
.C548 F0 15       BEQ $C55F
.C54A C9 06       CMP #$06
.C54C F0 1D       BEQ $C56B
.C54E C9 03       CMP #$03
.C550 F0 23       BEQ $C575
.C552 60          RTS
.C553 AD 8E 02    LDA $028E
.C556 29 04       AND #$04
.C558 D0 25       BNE $C57F
.C55A C6 FB       DEC $FB
.C55C C6 FC       DEC $FC
.C55E 60          RTS
.C55F AD 8E 02    LDA $028E
.C562 29 04       AND #$04
.C564 D0 23       BNE $C589
.C566 E6 FB       INC $FB
.C568 E6 FC       INC $FC
.C56A 60          RTS
.C56B AD 8E 02    LDA $028E
.C56E 29 04       AND #$04
.C570 D0 21       BNE $C593
.C572 C6 FC       DEC $FC
.C574 60          RTS
.C575 AD 8E 02    LDA $028E
.C578 29 04       AND #$04
.C57A D0 21       BNE $C59D
.C57C E6 FC       INC $FC
.C57E 60          RTS
.C57F A4 FD       LDY $FD
.C581 C8          INY
.C582 C0 40       CPY #$40
.C584 F0 02       BEQ $C588
.C586 84 FD       STY $FD
.C588 60          RTS
.C589 A4 FD       LDY $FD
.C58B 88          DEY
.C58C C0 FF       CPY #$FF
.C58E F0 02       BEQ $C592
.C590 84 FD       STY $FD
.C592 60          RTS
.C593 A4 FE       LDY $FE
.C595 C8          INY
.C596 C0 40       CPY #$40
.C598 F0 02       BEQ $C59C
.C59A 84 FE       STY $FE
.C59C 60          RTS
.C59D A4 FE       LDY $FE
.C59F 88          DEY
.C5A0 C0 FF       CPY #$FF
.C5A2 F0 02       BEQ $C5A6
.C5A4 84 FE       STY $FE
.C5A6 60          RTS
```

Disassemblato 2

```
.C200 A9 0D       LDA #$0D
.C202 8D 21 D0    STA $D021
.C205 AD 18 D0    LDA $D018
.C208 09 02       ORA #$02
.C20A 8D 18 D0    STA $D018
.C20D 4C 00 C5    JMP $C500


.C400 A9 06       LDA #$06
.C402 8D 21 D0    STA $D021
.C405 AD 18 D0    LDA $D018
.C408 29 FD       AND #$FD
.C40A 8D 18 D0    STA $D018
.C40D 4C 00 C5    JMP $C500
```

Disassemblato 3

```
.C03A A9 5C       LDA #$5C
.C03C 85 FE       STA $FE
.C03E A2 04       LDX #$04
.C040 A0 00       LDY #$00
.C042 84 FD       STY $FD
.C044 A9 0D       LDA #$0D
.C046 91 FD       STA ($FD),Y
.C048 C8          INY
.C049 D0 FB       BNE $C046
.C04B E6 FE       INC $FE
.C04D CA          DEX
.C04E D0 F6       BNE $C046
.C050 A2 20       LDX #$20
.C052 98          TYA
.C053 91 FD       STA ($FD),Y
.C055 C8          INY
.C056 D0 FB       BNE $C053
.C058 E6 FE       INC $FE
.C05A CA          DEX
.C05B D0 F6       BNE $C053
.C05D 85 33       STA $33
.C05F 85 37       STA $37
.C061 A9 5C       LDA #$5C
.C063 85 34       STA $34
.C065 85 38       STA $38
.C067 A9 3F       LDA #$3F
.C069 85 FD       STA $FD
.C06B 85 FE       STA $FE
.C06D 60          RTS


.C200 AD 02 DD    LDA $DD02
.C203 09 03       ORA #$03
.C205 8D 02 DD    STA $DD02
.C208 AD 00 DD    LDA $DD00
.C20B 29 FC       AND #$FC
.C20D 09 02       ORA #$02
.C20F 8D 00 DD    STA $DD00
.C212 AD 18 D0    LDA $D018
.C215 09 78       ORA #$78
.C217 8D 18 D0    STA $D018
.C21A AD 11 D0    LDA $D011
.C21D 29 7F       AND #$7F
.C21F 09 20       ORA #$20
.C221 8D 11 D0    STA $D011
.C224 4C 00 C5    JMP $C500


.C400 AD 02 DD    LDA $DD02
.C403 09 03       ORA #$03
.C405 8D 02 DD    STA $DD02
.C408 AD 00 DD    LDA $DD00
.C40B 29 FC       AND #$FC
.C40D 09 03       ORA #$03
.C40F 8D 00 DD    STA $DD00
.C412 AD 18 D0    LDA $D018
.C415 29 15       AND #$15
.C417 8D 18 D0    STA $D018
.C41A AD 11 D0    LDA $D011
.C41D 29 5F       AND #$5F
.C41F 8D 11 D0    STA $D011
.C422 4C 00 C5    JMP $C500
```

Disassemblato 1 - Ecco la versione di base del programma a parte pubblichiamo le aggiunte relative alla versione 1 (disassemblato 2) e alla versione 2 (disassemblato 3)
Disassemblato 2 - Ecco le aggiunte da fare al disassemblato 1 per ottenere le due pagine testo differenziate
Disassemblato 3 - Queste sono le modifiche da fare al disassemblato 1 per ottenere la mezza pagina in alta risoluzione

ta, che venga eseguita quella successiva, si ha quindi un lampeggiare piuttosto fastidioso. È questo il motivo per cui, senza precedenti modifiche, se si fa partire il programma si ottiene questo risultato, dal momento che l'inizio e la fine della finestra sono sullo stesso punto. Basta ovviamente premere F7 e la finestra si amplia, interrompendo l'effetto. Infine, per far partire il tutto, una volta fatto andare il programma in Basic si fa la SYS 49152, e per interromperne l'esecuzione si preme RUN/STOP e RESTORE.

Il programma di base viene utilizzato in duplice veste. La prima è semplicissima: all'interno della finestra il colore di sfondo cambia in verde chiaro, e il set di caratteri diventa minuscolo; può essere utile per evidenziare parti dello schermo, e comunque in genere per migliorarne la visualizzazione. La seconda versione serve per visualizzare all'interno della finestra l'alta risoluzione. Purtroppo a tale fine ho dovuto sacrificare i 17K finali di memoria (non si sarebbe potuto fare altrimenti, tranne l'esclusione seppur momentanea del sistema operativo): lo schermo in alta risoluzione, pulito e posto a color verde chiaro, inizia a 24576 — i punti accesi appariranno in nero (con ottima visibilità per l'utente, ndr) — e volendo cambiare questa disposizione i 1000 byte di schermo si trovano a partire da 23552. Poiché la locazione di inizio schermo non viene comunicata al sistema operativo (basta una semplice POKE nella locazione 648) si ottengono effetti degni di nota: le operazioni con il cursore, anche se nascosto sotto la finestra, vengono regolarmente eseguite sullo schermo normale che parte da 1024, e nella parte in alta risoluzione non appaiono i tipici quadrati colorati dovuti alla coincidenza tra la mappa del testo e quella dei colori in hires, ovvero i due schermi sono completamente indipendenti. Notate che per predisporre l'alta risoluzione bisogna agire sulla locazione 53265, che contiene anche l'ottavo bit del registro del raster: se quindi si esegue un'istruzione del tipo

```
LDA (53265)
ORA #32
STA (53265)
```

e l'MSB del raster è uguale a 1, verrà ovviamente riscritto nel bit 7 della locazione 53265, e considerato come valore di confronto del registro del raster, cosa che non deve succedere, perché se l'MSB vale 1 le operazioni sullo schermo si svolgono interamente al di fuori della zona visibile, nella mia routine preparatoria, infatti, ho provveduto ad azzerarlo. Quindi prima di STA bisogna eseguire un

```
AND #127
```

Completiamo l'ottimo lavoro del lettore con un programmino-ino-ino in Basic che mostra una sinusoide sulla pagina grafica, posta in alto sullo schermo, mentre una semplice didascalia rende visibile la mezza pagina testo posta inferiormente. Concludiamo invitando il lettore ad approfondire gli argomenti da lui citati nella lettera acclusa al programma e alla documentazione, dato il sicuro interesse che susciterebbero.

Paroliamo

di Davide Pitto - Genova

Il programma che vi invio gestisce il noto gioco TV "Paroliamo" usando il Basic del CBM 64. Lo svolgimento prevede di tirare a sorte dieci lettere, vocali (tasto V) o consonanti (tasto C), a scelta del giocatore di mano, e di costruire la parola più lunga possibile nel tempo massimo di 45 secondi. Le dieci lettere vengono visualizzate in grande formato nella parte centrale dello schermo. Il programma non è ovviamente in grado di determinare l'esistenza della parola scelta, ma semplicemente controlla l'appartenenza delle sue lettere alle dieci sorteggiate, e sceglie la vincente tra quelle dei due giocatori, ovviamente vince la più lunga, e in caso di parità ha la meglio il giocatore di mano; in ogni caso, il punteggio è dato dal numero di lettere della parola. Le manche previste sono 8, e per ognuna i punti ottenuti vengono sommati ai precedenti, segue la proclamazione del vincitore.

La caratteristica fondamentale del gioco è però la possibilità di giocare contro il computer, semplicemente digitando COMPUTER alla richiesta del nome di uno dei giocatori; in tal caso la scelta delle dieci lettere avverrà automaticamente. Il sistema si basa su quasi 1000 parole, memorizzate nelle linee DATA sotto forma cifrata per impedire qualsiasi forma di spionaggio, la cui lunghezza varia da 4 a 10 lettere, che verranno usate in proporzione al livello di difficoltà selezionato. Questo varia da 1 a 9 con difficoltà crescente, mentre il livello 0 (zero) si comporta un po' a caso, quasi in modo umano. Le parole codificate nei DATA sono leggibili tramite la routine posta nelle linee 30000 e seguenti, che quindi non sono di immediato uso nel corso del programma. Qualora poi si volesse limitare il gioco solo a due esseri umani — senza estenderlo al computer — si potranno tranquillamente omettere le linee DATA, di gran lunga la parte più onerosa da digitare. Per i più pigri c'è sempre la cassetta di MC.

(segue da pagina 135)

3

```
1160 E=1
1200 I=1 FORL=1TO10
1210 L$(1)=MID$(E$,L 1)
1220 IFI<10THENI=I+1
1230 NEXT
1300 FORL=1TOL2 I=0
1310 I=I+1 IFMID$(PS$ L,1)=L$(1)THENL$(I)="" GOTO1330
1320 IFI<10THEN1310
1330 NEXT
1340 C$=L$(1)+L$(2)+L$(3)+L$(4)+L$(5)+L$(6)+L$(7)+L$(8)+L$(9)+L$(10)
1350 IFLEN(C$)=10-L2THENPRINT "LA PAROLA DI "NS$" E' ESATTA" GOTO1500
1360 E=E+2
1500 IFE=1THENPRINT "LA PAROLA DI "NP$" E' ERRATA"
1550 IFE=1THENPRINT "VINCE "NS$" CON"L2"LETTERE" SS=L2 GOTO2020
1600 IFE=2THENPRINT "LA PAROLA DI "NS$" E' ERRATA"
1650 IFE=2THENPRINT "VINCE "NP$" CON"L1"LETTERE" SP=L1 GOTO2020
1700 IFE=3THENPRINT "ENTRAMBE LE PAROLE SONO ERRATE"
1750 IFE<>3THEN2000
1800 PRINT "NON PRENDETE NESSUN PUNTO" GOTO2100
2000 IFL1>=L2THENSP=L1 PRINT "VINCE "NP$" CON"L1"LETTERE" GOTO2020
2010 SS=L2 PRINT "VINCE "NS$" CON"L2"LETTERE"
2020 IFN/2=INT(N/2)THENS1=S1+SS S2=S2+SP GOTO2100
2030 S1=S1+SP S2=S2+SS
2100 FORW=1TO10 GETPU$ NEXT
2200 PRINT "PREMERE UN TASTO PER CONTINUARE"
2300 GETK$ IFK$=""THEN2300
2350 IFCOTHENTA=TA+1
2400 NEXT
2500 IFS1>S2THENPRINT "VINCE "N1$" PER"S1"A"S2 GOTO3000
2600 IFS1=S2THENPRINT "PARITA !!!"S1"A"S1 GOTO3000
2700 PRINT "VINCE "N2$" PER"S2"A"S1
3000 PRINT "UN ALTRA PARTITA (S/N) ?"
3001 GETR$ IFR$<>"S"ANDR$<>"N"THEN3001
3100 IFR$="S"THEN3
3200 PRINT"                    "TAB(14)"**** CIAO ****" END
5000 A=(LD%-1)/2 B=(LD%/2)-1
5020 IF(LD%/2)=INT(LD%/2)THEN5040
5030 VO=INT(RND(0)*(NT-NP*((9-LD%)/2)-NP*A)+NP*A+1) GOTO5100
5040 IFLD%=0THEN5060
5050 VO=INT(RND(0)*(NT-NP*(8-LD%)/2)-NP*B)+NP*B+1) GOTO5100
5060 VO=INT(RND(0)*NT+1)
5100 LU=LEN(PA$(VO))
5110 FORI=1TOLU
5115 AS=ASC(MID$(PA$(VO),I,1)) IFAS=65THENAS=91
5120 LE$(I)=CHR$(AS-1) NEXT
5200 IFLU=10THEN5300
5210 AL$="AEIOUABCDEFGHILMNOPQRSTUVZAEIOU"
5220 FORI=LU+1TO10
5230 TR=INT(RND(0)*31+1)
5240 LE$(1)=MID$(AL$,TR,1) NEXT
5300 LE$=LE$(1)+LE$(2)+LE$(3)+LE$(4)+LE$(5)+LE$(6)+LE$(7)+LE$(8)+LE$(9)+LE$(10)
5305 PS$=LEFT$(LE$,LU)
5310 FORI=1TO10
5320 SC=INT(RND(0)*10+1) MI$=MID$(LE$,SC,1)
5330 IFMI$="@"THEN5320
5340 LE$(I)=MI$ LE$=LEFT$(LE$,SC-1)+"@"+RIGHT$(LE$,10-SC) NEXT
5400 FORI=1TO10 CH=(ASC(LE$(I))-64) C1=(ASC(LE$(I))-75)
5410 IFASC(LE$(1))>75THEN5440
5420 ONCHGOSUB100,150,160,170,110,180,190,200,120 NEXT
5430 RETURN
5440 ONC1GOSUB210,220,230,130,240,250,260,270,280,140,290,5460,5460,5460,300
5450 NEXT
5460 RETURN
6000 DATABOUB BOTB BODB BUFP BSJB
6010 DATABHJP BSUP BSUF CBSB CFSF
6020 DATACJSP CBSP CFOF CVOP DBSP
6030 DATADVSB DFSP DVUF DBTP DVCP
6040 DATADBMP DBPT EBSF EJSF EBUP
6050 DATAEPUF EJUP EPTF EJWP FSUB
6060 DATABMCB OPED DBWP EVDB CBWB
6070 DATAGBSP GJMP GFUP GBUP GPUP
6080 DATAGBTF GVHP GBNF GJMN HBSB
6090 DATAHJSP HVOP HPUB HJDB HBMB
6100 DATAHPMB HFMP I-UP INFP IDFB
```

4

```
9270 DATANPOBMF RBHBSB APUJOP DBOVUP DPOBUP
9280 DATAMFOJEP MVOJEP
9000 DATABOJNBMF BOUFOOB BSBODJB BTQFUUP BTTJPNB
9010 DATABTTVOUP BTUSVTP BUUJHVP CBUUVUP CJWBODP
9020 DATACFMMJDP CJOPNJP CJTPDUF CPMMJSF,CPTTPMP
9030 DATAD1JDBAB D1MFDCB DJOHJJB,DJMJDJP D1NJFSP
9040 DATADFNBDEP DFNQJUP DPOWJUP EFOFOUF EFDPFUP
9050 DATAEFEVSJF EFMJAJB EFMGJDP EFQPSSF EPWJAJB
9060 DATAFFOLSBUB FTVJBSF FOJMPHP FSFNJUB FSPEFSF
9070 DATAFVOPSJB FWPMVUP GBSJOHF GBGGBSB GBWFMMB
9080 DATAFBUUVSB GPEPOEP CFSSBSF GJDDBSF HVBSJSF
9090 DATAHFJJFMMP HJVEJOF HJPWBSF HJVCJMF HJVONRR
9100 DATAHPOEPMB JEJFAJB JEJMMJP DJTHMJB JNJFTTP
9110 DATAJNOVNTP JOBMBSF JODBOUP MBNVBEB MBNCJSF
9120 DATAMBNBDSP MFNHFSF NFUBSHP MJRVPSF MJFWJUP
9130 DATAMVDFSOB HBC1FSB NBUSPOB NBUSJDF NFMFOTP
9140 DATANFSFOEB NFSDFEF NJFUFSF HJOVAJB NVOHFSF
9150 DATAOPNFMMB OP IRSJP DPRJPOF UVUSJSF OVUSJUP
9160 DATAPPFNBLP PCMJPUP PCMVOHP FDVMBUP PHHFUUP
9170 DATAPNJDJEB POPTBAF PSEJHOP OBSMBSF OBSPEJB
9180 DATAQBCNJSF QBTTBUP QBTDFSF QFTDBSF QJTFMMP
9190 DATAQHBDFSF QMBHJOP RVBUUSP PVFSFMB PVFSDJB
9200 DATAFVBMDIF FVBMPSB SBEJBUP SBHSAAP SFBHJSF
9210 DATASFHHF F SFHJPOF SJ FMMF SFEVBSF SVNHJUP
9220 DATATDBUPMF TDBWEHP TDFHR1P TFDDBSF TFDDIJP
9230 DATATFNJFUP TFHVBDF TJDBSJP TJNOFSF TUFTVSB
9240 DATAUBNOPOF UBSJUGB UBUVBSF UPSDFOF UPSDIJP
9250 DATAUPTTJDF UVGGBSF VOBOJNF VTUJPOF VOAJPOF
9260 DATAHOMBOHB WBNOJ-P WFOUPMB WFDUVSB WFSCBMF
9270 DATAAFOAFSP ABAAFJB,AFSCJOP ABOABSB APEJBDP
9280 DATAHBNFS B ATOHBSP
10000 DATABOHVT JB ROP 1TUP BMJRVPUB BGGJEBSF BMOBCFUP
10010 DATABMMFWBUP BMMPNJJP BHDAAPOF CJAMBSSP FJSCBOUF
10020 DATAC1D1OJJF CJGSPOUF CPUBOJDB CSBOEJSF CVHJBSEP
10030 DATADDBJUBSF DBBSJPMB DBSDJPGP D1JVEFSF D1JPTBSF
10040 DATADIJFEFSF DIJTVSNP D1JFSJOB DIJUBSSB EFMVEFSF
10050 DATAEFUPCBSF EFUSBSSF EJBNFUSP FJBMFUUP EJTHVTUP
10060 DATAECJUBSF FMFHBOUF FMFUUPSF FHFSHFSF FN1HSBUP
10070 DATAFOJHSBOF FOJEFNJB FSHFUJDP FSVAJPOF GBNJHMJB
10080 DATAUB EFMMP OF HFOUP G1UUJAJP GJBDDPMB GJFSFAAB
10090 DATAGMBHFMMP GMFUUFSF HFTJJPOF HJJEDDJP HJBD1OUP
10100 DATAHMPSJPTP HPNJLPMP HVBSEBSF HVBTUBUP HVJAABSF
10110 DATAJFFDHJDP JHOPCJMF JHOPSBUP JNNBNJOF JNOSPOUB
10120 DATAJJDB DP JAHJVCJB MFOAVPMP MFPOBSEP MJCFSBMF
10130 DATAMJODJEUP MPHPSSFB MVDJEBSF MVTTVS1B NBONJBUP
10140 DATANBOJHMJB NFNCSBDB NFUAJPOF NFJLJJOP NPTH BSF
10150 DATANVSBHMJB OFHMFUUP DPWFNCSF OBDD1FSB PSFDDIJP
10160 DATAPSHPHMJP PSSJCJMF PTOVSBUP PTT1HPOP PTUSV1SF
10170 DATAPUUVSBSF QBMFTUSB QBMOFCSB QBSUFOAB QBTTJPOF
10180 DATAQBTUSBOP QFEFTUSF QFEBHHJP QFSJDPMP QJFUJTNP
10190 DATAROPRJUJDB QSFOFUUP QSFTUBSF RVBEBSF RVBEFSOP
10200 DATARVBESBUP RVBSBDUB,RVJFTBSF SBODPMUB RVBUFSDB
10210 DATASBOTPEJB SFEJNFSF SFHJTUSP SFMBUJWP SJDFWVUP
10220 DATATBMVUBSF TBMWFAAB TBDAJPOF TOBHEBUP TDFOEFSF
10230 DATATDFOFSUB TFDUFOAB TFSHJHJP TPSCFUUP UPSNFOUP
10240 DATAU-BGVDUB USBUUBSF USBAJPDF USPUUPMB UVMJOBAP
10250 DATAVUFCTJMF VOJWFSTP VOJOPSNF VNJMJBSF VOHVFOUP
10260 DATAWFOEFUUB WFMJWPMP WFSSFUUP WFSHPNOB WJPMFOUP
10270 DATAWJPUUPMP WJTDPOUF WPNJUBSF WPMVCJMF ABNOPHOB
10280 DATAA1HCFMMP AVCFMBSF
11000 DATABLCFMMJSF BCDBEFNJB BCPSJHFOP BMBCBTUSP BOFTUFTJB
11010 DATABOOSFEBUP BTTBTTJDP BWBOJSFOP CFOFGJDJP CJTCJHMJP
11020 DATACNBUFSBSF CPNOSFTTP CPUUJHM1B CBVMJDBSF CVDJOUPSP
11030 DATADBNRBUVSB DBNPNJMMB DBNVGBSF DBOEFHHJP DIJBSFAAB
11040 DATADJDBUSJDF DJDMBNJOP DPODPSEJB EBWBOABMF EFCFMMBUP
11050 DATAEFDJTJPOF EJMJHFOUF EJMFHVBUP EJMVHJBSF EPQOJFARB
11060 DATAFEJCJDBSF FGGVTJPOF FMBCPSBUP FMFNPTJDB FPVJOPNTP
11070 DATAFSFE1UBSF FJHDTUPMP GBODJVMMP GBJBCVUUP GBTDJDPMP
11080 DATAGFDPOEJUB GJDDBOBTP GJFEEFAAB GGFRVFOUP HBMM1BSEP
11090 DATAHFNTJNJOP HFSNJOBSF HPWFSOBUP HSBU1OPNB HJFNC1VMF
11100 DATAHVBOOJBMF JMMVTJPOF JNCBMMBUP JNOBSUJJF JODFOOEUP
11110 DATAJODFOUJNP JTUFSJTNP JTUPSJBUP MBITSTJNP MBUFJAJF
11120 DATAMFHJJUJNP MJTFWJUBSF MJHOBHHJP DJCHJJBMF JNCSPHM1P
11130 DATANBESJHBMF NBEPNDBMF NFOEJDBSF HJBHPMB F HJJTDPMP
```

```
6110 DATABSTP GPSP DFOB AFMP MJEP
6120 DATAMBHP MJTP MFCP MVQP MPUP
6130 DATAMJNP MBNB MBSF NFMB N8MF
6140 DATANJUF NPUP NVTP NVDP N8HP
6150 DATAOFSP,OVDB OBTP OFWF OBOP
6160 DATAOVEP,OJEP OBWF,PSUP PU3F
6170 DATAPTUF PSTP PAJP PHHJ PTTP
6180 DATAQBMP QFMP QJMB QPDP Q8HP
6190 DATAQVSP QVNB QFTP BMDF BTUB
6200 DATAMBUP MFDB MFWB GJDF GVOF
6210 DATASBSP SFTP SJTP SVHB SPHP
6220 DATASBNP SFMP TFOP TJUP TVHP
6230 DATATBMF TFUF TFNF,TBOP UVUB
6240 DATAUVCP UBDB UFTP UJSP UPSP
6250 DATAUPOP UPQP VSOB WBTP WJDP
6260 DATAWFSP WPUP,WBHP WFMP WJTP
6270 DATAAFSP USBH,TFHP,OPUP SJUP
6280 DATASFUF GJEP
7000 DATABHJSF,BORVB,AFQQB BNBUP BEEJP
7010 DATABFSFP BNPSF BWWSP,CVPOP CFPOF
7020 DATACBJUB CSFWF CJWJP CPOUB CVTUB
7030 DATACBSDB DBSUB DPOUF,DJOUB DPNQB
7040 DATADPSUP,DSJOF DBSEP OIFUP EPUUP
7050 DATAEJUUB EFOUF EFOTP EBSEP EBOAB
7060 DATAFGJDP FSJDB FEFSB,FTVMF FHVMP
7070 DATAFEVMF AJODP FEJUP GJSNB GVSJB
7080 DATAOVSUP,OJTTP OBOHP GVOHP GBTUP
7090 DATAOJOUP HVTUP HFOUF,HFSMB NIJTB
7100 DATAHSVNP HFOJP HSFWF HPUUB JSJEF
7110 DATAJSPTP JSBUP JOWJP JOPOB FUFSF
7120 DATABWFSF MFOUF MVDSP MFAAP MBHGP
7130 DATAMJODF MJUSP MPSEP MFOSF HBAAP
7140 DATAHFAAP HBSAP NJTUA MVOUP HPDUF
7150 DATANITSUP HBSDB QFUUP OFNCP,DFSCP
7160 DATAOFSWP MJNCP OFTUB OBOUF PSBSF
7170 DATAPTBSF PDPSF PGFSB PEPSF BSBSF
7180 DATABSSUP OBTTP OJOAB OBAAP OVAAP
7190 DATAOJAAP OSFEB QPDUF QFSTP RVBTJ
7200 DATARVBMF RVPUP,OVPSF BMUSP HFOUP
7210 DATAFUJDP,SFTUP SVUUP SBDHP SJUUP
7220 DATASPNCP,SPUUP CPBUP CBVMF,SBAAP
7230 DATAAPMMB TFSOF TFHOA TFMWB TBCUP
7240 DATATPBWF TFSJP TPHOP,UPSWP UFSAP
7250 DATAAFDDA AJUUP UVSQF,UFQQB UJOUB
7260 DATAVEJSF VNPSF,VTBSF VEJUP,VNJEP
7270 DATAWBTUP,WJTUB WFDUP,WPMUP WJOUP
7280 DATAABCOB,APQQP
8000 DATABSEJSF BWPSJP BTTPMP BMABSF BRVJMB
8010 DATABVMJDP,BJVPMB CBTBSF CBEJMF CJEPOF
8020 DATALBCBMF CFWVUP CSVUUP CFTUJB CJBODP
8030 DATADI'FTB DIJBWF DPMBSF DPW8UP DPWBSF
8040 DATADBEFSF DFEFSF DBEVDP OFEVUP EPUBSF
8050 DATAEPOBUP,ERUPSF EPDJMF EFEJDB EFCJUP
8060 DATAEFTUSB FEPUUP FMPHJP,FOPSNF FSBSJP
8070 DATAFSFUUP FWFOUP FSQJDF GBCCSP GBDDJB
8080 DATAGBWPMB GJBODP GJCCJB GJHMJP GJMBSF
8090 DATAGVHBSF,GVNBUP HFMPTP HJVCCB HJVOUP
8100 DATAHSBUJT HSBAJB HVBJSF JCSJEP JMMFTP
8110 DATAJHIFOF,JNOFUP JDDBWP JOEJSF JOEBOP
8120 DATAMBCCSP M2NVOB MBVSFB MFEFSF MFOJSF
8130 DATAMFWBSF MFWBUP MVSJEP HBHMJP NBJBMF
8140 DATANJDDJB HJTVSB NPUJWP NVUBSF NVTJOB
8150 DATAOBTUSP OPUBSF OPUBJP DJUJEP OBUVSB
8160 DATAOJUPSF OVCJMF PUUJDP MJWJEP PSUJOB
8170 DPTABMABUP,PDDIJP PSOBUP PSEJUP QBHBSF
8180 DATAOPUFSP,OMMJSF QBWPOF QVOJUP QVEJDP
8190 DATAQPUBSF QPTBUP RVBESP RVBOUP RVFTUP
8200 DATAPVFMMP RVJFUP RVBSAP RVBSUP RVJOUP
8210 DATASPTUSP SBEFSF SJEFSF SVWJEP SFNBSF
8220 DATACBOBUP TBORUP TJMVSP TBKBDF TJNJMF
8230 DATATBOVUP TPOJUP,UFOVUP UPTBSF,UVSBSF
8240 DATAUPOBJB UFMPOF UJHJEP YSUBSF VOIPOF
8250 DATAVSMBUP EJWBOP YSUBUP YDH1JB WPUJWP
8260 DATAWPMOUP,WBMFSF WJHJMF,WJOJOP WBJPMP
11140 DATANPOVNFOUP NVDJAJPOF OFHPAJBSF OFDDIJFSP OPTPDPNJP
11150 DATAPCJFUUBSF PCCSPCSJP,PDDBTJPOF PDDIEFOUF PMPOBVUP
11160 DATAPNJTTJPOF PSDIFTUSB PSJAAPOUF OBMJAABUB OBSFOUFTJ
11170 DATAQBUUVHMJB QBWJNFOUP QFDVMJBSF QFMMJOBOP QPOUFOJOF
11180 DATAQSPHSBNNB QSPTDFDJP QVUJOFSJP RVBSUJFSF RVPAJFOUF
11190 DATAKVBSFTJNB RVFTUJPOF RVBUUSJOP SBDDIFUUB SBEJDDIJP
11200 DATASBGGJOBUP SBHJPOBSF SFJUFSBSF SFMJHJPOF,SJOBNOBUP
11210 DATATBOTDBJUP TBOUVBSJP TCBHMJBSF TDBNCJBUP,TDIFABSF
11220 DATATDIFSDJUP TDIFMFUSP TGPMHPSJP TFOTJCJMF TUSBAJBSF
11230 DATATUSJDHFSF TUSJMMBSF TUSPODBUP TVDDIJBSF UFHYNFOUP
11240 DATAUFHQFSBUP UPSQFEJOF UPMMFSBSF USBNCVTUP USBTBMJSF
11250 DATAVDDJTJPOF VMUFSJPSF VOJOJOBUP VTJHDVPMP,WBOJMMBSF
11260 DATAWBHBCPOEP WBMFSJBDB WBOUBHHJP WBSJDFNMB WFHMJBSEP
11270 DATAWFOUJMBUP WFMPESPNP ABOOFSBOP AJCBMEPOF AJCFNMJOP
11280 DATAWJSVMFOUP USBQFMBUP
12000 DATABDRVITUBSF BCPMJAJPOF BDDBTDJBUP BDVUJAABSF BOPVJJJOP
12010 DATABDOVTBUPSF BOBMOBCFUB BOBMJABSF,BWWFOUJAJP CBMBHMJBJP
12020 DATACBSOPMMBSF CFSQPOPMF CPSTBJVPMP CVSSBTOPTP CSFWFUUBUP
12030 DATADBN2PTBOUP DBWBMMFSJB DIJPDDJPMB,DJPODPMBUB DPDDJDFMMB
12040 DATADGJTBOUFNP DSPDFGJTTP EFCJMJUBUP EFGSBVEBSF EFNPOSBAJB
12050 DATAEJTUJDUJWP ESBNNBUJDP EPQPHVFSSB FDIFMMJBSF FEVDBAJPOF
12060 DATAFOJMFUUJDP FTUJOHVFSF FWPMVAJPOF FTUSJDTFOP DBCCJTPMOP
12070 DATAGBMDJEJBSF GMVUUVBOUF OSBDBTTBUP GSFTDIFAAB GSVUUJGFSP
12080 DATAOPOEBAJPOF HFOFBMPHJB HJOOBTUJDB HJVEJAJPTP HPODJPNBSF
12090 DATAHSBWJEBOAB HVBSJHJPOF HVBEBHOBSF JMMVHJOBUP JNQBODJBJP
12100 DATAJODPHMJFSF JDEVMHFOAB JOOSBAJPOF JOPDPOESJB JUFSBAJPOF
12110 DATAMBTUSJDBSF MFOUJDDIJB MJOHVSHHJP MPDPHPUJWB MPNCBHHJDF
12120 DATAMVTTBAJPOF MVTTVSJPTP NBNNFBCJMF NBHHJPMJOP NBTTFSJAJF
12130 DATANBTDIFSBUP HJTFSBCJMF NJOBDDJBSF NPSJHFSBUP NVMUJGPSNF
12140 DATAHPTDIJDJEB,DBVDSBNDSF OFHMJHFOUF DPCJMJUBUP DVCJGSBHJP
12150 DATAOPMFHHJBUP PCCFEJFOUF POEFHHJBSF POF3BAJPOF PTUSBDJJNP
12160 DATAPSJHJOBSJP PTTFRVJBSF QBESPOBOAB OBHMJBDDJP,QBSBHPOBSF
12170 DATAQBDFHJSJDP QBSDIFNNJP,QFHHJPSBUP,QFTTJNJTUB QJUUPSFTDP
12180 DATAOPSUJCFSJB QSFGBAJPOF APNGBOFMMP QSFTJEJBUP RVBDUVORVF
12190 DATARVFSFMBOUF RVFTUJSJOF RVPUBAJPOF RVPUJEJBOP SBDDPSEBUP
12200 DATASJOGPSABSF SBNNFOUBSF SBQJOBUPSF SFOJWJFOUF SFOVDDJMJDB
12210 DATASFQFSUBSJP SJOUVAABSF TBMWBHFDUF TBDAJPOBSF TBDDIJDJDJP
12220 DATATOBDEBHMJP TDBSOFSBSF TOJBDRVBSF,TDJFOAJBUP TDPNQJHMJP
12230 DATATDSPTDJBSF TNBDDIJAUP,TPHHFAJPOF TUBSOVUJSF TUBUJTUJDB
12240 DATAUBQOFAABUP UBTTBAJPOF UFNOFTUPTP UFOUFOOBSF UPSDJOPMMP
12250 DATAUSBWFSTBSF USJCVUBSJP USJWJBMJUB USPWBUFMMP VCJDBAJPOF
12260 DATAVHVBHMJBUP VNBOJUBSJP VUJMJUBSJP WBDJMMBOUF WBMMJHJBOP
12270 DATAABN2JMMBSF WBSJBAJPOF WFMFHHJBSF,WFSFDPOEJB WFSHPHOPTP
12280 DATAWJTJCJMJUB WPMVUUVPTP *
30000 PRINT"J" FORC=1TO994
31000 READPR$ IFLEN(PR$)>(C/142+4)THENPRINTPR$ STOP
32000 FORA=1TOLEN(PR$)
33000 AS-ASC(MID$(PR$,A 1))
33500 IFAS=65THENAS=91
34000 L$(A)-CHR$(AS 1) NEXTA
35000 PR$=L$(1)+L$(2)+L$(3)+L$(4)+L$(5)+L$(6)+L$(7)+L$(8)+L$(9)+L$(10)
36000 PRINTPR$ FORI=1TO10 L$(I)="" NEXTI NEXTC
40000 REM DAVIDE PITTO / 16139 GE
40001 REM CSO DE STEFANIS 25/20
40002 REM TEL, 010 / 873270
```

Questo programma è disponibile su cassetta e disco presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 157

MI.PE.CO. VENDITA PER CORRISPONDENZA

ESTENSIONE PER SPECTRUM
299.000 tutto compreso 3 mesi di garanzia

COMPRESE 4 cartucce con 5 programmi (introductory, master file, tasword two antattack, games designer), **istruzioni in italiano**

INTERFACCIA UNO

MICRODRIVE

SPECIALE 8 CARTUCCE PER MICRODRIVE 49.000

SPECTRUM 48K PLUS

con lo SPECTRUM plus **manuale in italiano** *e in regalo 5 programmi in italiano (conto corrente, grafica funzioni, bioritmi, esapedone + il Supercopiatore di Massimo Rossi)*

QL 1.199.000
alimentatore manuale in inglese 12 cartucce con 14 programmi

nuovo SPECTRUM 48K + ... 339.000
manuale in italiano cavetti alimentatore cassetta dimostrative e oltre 50 000 lire di software originale in italiano

SPECTRUM 48K 339.000
manuale in italiano cavetti alimentatore cassetta dimostrative e oltre 50 000 lire di software originale in italiano

INTERFACCIA UNO + MICRODRIVE 299.000
(4 cartucce con 5 programmi masterfile tasword two ant attack games designers e cartuccia dimostrativa)

MICRODRIVE 149.000
si usa con l interfaccia uno
Compresa 1 cartuccia con programma dimostrativo

STAMPANTE ALPHACOM 32 .... 199.000
per Spectrum e ZX 81 istruzioni in italiano 1 rullo di carta in regalo

EPSON RX 80 F/T + ... 769.000
con interfaccia Centronics

AVVERTENZE:

- tutti i prezzi sono comprensivi di IVA e spese postali
- -pagamento contrassegno al ricevimento del pacco
- -segreteria telefonica in funzione fuori orario, chiedete listini o altre informazioni Vi risponderemo
- **-sconti quantità**

INTERFACCIA PARLANTE CURRAH 99.000
manuale completo in italiano Tutti i suoni attraverso il Vostro televisore

TRISLOT 27.000
presa triple per Spectrum

10 RULLI di carta termica × ALPHACOM 32 .... 39.000

8 CARTUCCE × MICRODRIVE 49.000

ESPANSIONE + 32K × SPECTRUM .. 79.000
issue 2 o 3 specificare facilissime da montare istruzioni dettagliate in italiano con fotografie porta il Vs Spectrum da 16K a 48K (ad esaurimento)

SUPER COPIATORE con verify 15.000
può caricare più blocchi con senza o con testata falsa e li riversa tutti insieme originale in italiano

TASTIERA DELLO SPECTRUM PLUS 79.000
Kit per trasformare lo Spectrum normale in Plus (liste di attesa)

NIKE BACK UP 79.000
mantiene l alimentazione allo Spectrum per oltre 30 minuti anche se viene a mancare la tensione di rete Batterie nickel cadmio comprese

PARTI DI RICAMBIO PER SPECTRUM

GARANZIA 48H
la MI PE CO si impegna a sostituire tutto il materiale spedito se trovato malfunzionante entro 48 ore dal ricevimento

INFORMAZIONI E ORDINI:

MI.PE.CO. - Cas. Postale 3016 00121 ROMA (OSTIA) - Tel. 06/5611251

Anche questo mese abbiamo una rubrica nutrita: prima di partire con i programmi vogliamo però rivolgervi una preghiera. Per cortesia, se decidete di mandare qualcuno dei vostri lavori in redazione, scrivete la lettera, l'articolo di accompagnamento, e tutto quello che altro volete, con la macchina da scrivere e NON con il Tassword II in unione alla ZX printer!!

Abbiate pietà della nostra povera vista, già affaticata dalle troppe ore trascorse davanti a monitor e TV di tutti i tipi...

Va da sé che anche l'Alphacom o la Seikosha GP 50 sono da evitare a tutti i costi, mentre una stampante normale da 80 colonne va benissimo, se proprio non sapete come fare, sempre meglio la normale stilografica (scrivere chiaramente, per favore) che degli obbrobri a 64 colonne in caratteri formato Lilliput. Finita la supplica, possiamo incominciare.

Instring/Rubrica telefonica ON ERROR GOTO

di Francesco Balena - Bari

Vi invio tre programmi per la rubrica dedicata al software dello Spectrum. Più precisamente si tratta di:

1) una routine in linguaggio macchina per implementare la funzione INSTRING (o SUBSTRING), assente nel Basic Sinclair.

2) RUBRICA TELEFONICA, un semplice archivio che utilizza la precedente routine per la ricerca veloce dei dati.

3) Una routine in linguaggio macchina che simula l'istruzione ON ERROR GOTO (e ON ERROR GOSUB), anch'essa assente sullo Spectrum.

Instring

Una delle funzioni tradizionalmente assenti nel Basic degli home ed i personal economici è la INSTRING, che serve a verificare se una certa stringa è contenuta in un'altra.

Tale assenza è in parte giustificata dal fatto che è possibile simularla in Basic, ma nel caso particolare del Basic Sinclair, di cui è proverbialmente nota la lentezza, si tratta di una soluzione inaccettabile nella maggior parte dei casi.

La routine in linguaggio macchina che vi presento rimedia a questa mancanza ed è ovviamente velocissima: non è molto lunga e, cosa fondamentale, è perfettamente rilocabile.

Non si deve fare altro che introdurre il programma di caricamento esadecimale (listato 1) e dare il RUN, se non ci sono errori apparirà il messaggio "Start Tape..." e potrete salvare su nastro la routine sotto forma di file di byte, per ricaricarla in un secondo momento ed utilizzarla nei vostri programmi.

La funzione INSTRING ha di solito la seguente sintassi:

INSTRING (a$,b$)

e restituisce un numero intero positivo che indica la posizione della stringa b$ all'interno della stringa a$, oppure zero se a$ non contiene b$.

Così, ad esempio

INSTRING ("MCmicrocomputer","com")

restituisce il valore 8, mentre

INSTRING ("MCmicrocomputer","COM")

restituisce il valore 0.

Esiste anche una versione più completa della stessa istruzione:

INSTRING (a$,b$,n)

che ricerca la stringa b$ a partire dal carattere ennesimo di a$, ed è molto utile quando la sottostringa è contenuta più di una volta nella stringa principale.

Ad esempio

INSTRING ("una casa rosa", "sa",1) = 7

mentre

INSTRING ("una casa rosa","sa",8) = 12

la funzione di INSTRING che ho preparato è del tipo completo, a tre parametri, anche se la sua sintassi è leggermente più complessa:

LEN (a$ + (b$ AND 8 X USR ins))

dove "ins" rappresenta l'indirizzo al quale abbiamo collocato la nostra routine, che, lo ricordiamo, è rilocabile.

Ad esempio possiamo scrivere

LEN ("una casa rosa" + ("sa" AND 8 X USR ins))

Listato 1

```
   1 REM ************************
   2 REM *                      *
   3 AEM *       INSTRING       *
   4 AEM *      HEX LOADEA      *
   5 REM *                      *
   6 AEM ************************
   7 REM
  20 DEF FN h(b$)=CODE b$-48-7*(
CODE b$>57)
  30 CLEAA 31999  LET add=32000
  40 FOR k=100 TO 190 STEP 10  A
EAD a$  AEAD sum
  50 FOA j=1 TO LEN a$ STEP 2
  60 LET byte=16*FN h(a$(j))+FN
h(a$(j+1))
  70 POKE add,byte  LET add=add+
1
  80 LET sum=sum-byte  NEXT j
  85 IF sum<>0 THEN PAINT "EAAOA
E IN AIGA ",k  BEEP 1,30  STOP
  90 NEXT k
```

```
  95 SAVE "instring"CODE 32000,1
12
 100 DATA "CD991E7881CA202A0BED4
3A15CCDF128",2018
 110 DATA "ED43A75CED53A55CCDF12
8E822A35CED",2390
 120 DATA "58A75C7A8320050100001
82AED5BA15C",1336
 130 DATA "E8C5E3C1A7ED423005010
0001818444D",1569
 140 DATA "788128122AA15C19ED5BA
55C1ABE2810",1532
 150 DATA "230B788120F2CD822ACD8
22AEFA038C9",2123
 160 DATA "C5E5E52AA78CE3E3287CB
5E328100878",2172
 170 DATA "81281A13231ABE28EEE1E
1C118D2E1E1",2118
 180 DATA "C1ED5BA35CA7ED5223444
D18C9E1E1E10100",2343
 190 DATA "0018C1",217
```

Disassemblato della routine Instring

che restituisce 12, come nel caso precedente.

Se una delle due stringhe è di lunghezza nulla, oppure se il valore di n e superiore alla lunghezza della prima stringa, la funzione dà come risultato zero, mentre se n è nullo oppure negativo da errore.

Se questa sintassi vi sembra un po' troppo complicata potete definire una funzione del tipo

DEF FN i(a$,b$,n) = LEN (a$ + (b$ AND n ≤ USR ins))

oppure, se non vi interessa la forma completa

DEF FN i(a$,b$) = LEN(a$ + (b$ AND 1 ≤ USR ins))

Si noti che il parametro n va sempre specificato, anche quando è uguale a 1

Rubrica telefonica

Per chi si chiede a che cosa possa servire, in pratica, la routine di INSTRING sopra presentata, ecco subito una prima applicazione: una comoda rubrica telefonica.

Digitate il listato 2, date RUN 900 e poi caricate da cassetta la routine di INSTRING. Sullo schermo apparirà la lista delle opzioni disponibili, che potranno essere richiamate premendo l'iniziale del nome.

Le quattro opzioni sono:

1) Inserisci
2) Ricerca
3) Lista
4) Cancella

Per tornare al menu principale è sufficiente premere ENTER come risposta alla richiesta di input.

Scegliendo l'opzione 1 occorre scrivere nome, cognome e numero di telefono in un unico rigo, separandoli con uno o più spazi.

L'opzione 2 è, ovviamente, quella più interessante. La ricerca è pressoché istantanea, come chiave si può fornire una stringa qualsiasi, verranno evidenziati tutti i nominativi che la contengono.

Si può usare questa caratteristica per rintracciare dei numeri che si ricordano solo parzialmente, oppure tutti i cognomi che cominciano con una determinata lettera (basta specificare la lettera preceduta da uno spazio).

L'opzione 4 funziona come quella appena vista, con la differenza che per ogni nominativo ci viene chiesta conferma della volontà di cancellarlo o meno. L'opzione 3 è la più semplice: viene stampato il contenuto della rubrica, in ordine di introduzione.

Per uscire dal programma si deve premere il tasto 0, in questo caso si ha la possibilità di salvarlo assieme ai dati ed alla routine di INSTRING; la volta successiva il programma andrà in autostart e provvederà a caricare automaticamente i byte del linguaggio macchina.

I nominativi della rubrica sono contenuti tutti in un'unica stringa a$, separati fra loro da un carattere di controllo. Questa soluzione, rispetto a quella solita che consiste nell'utilizzare una matrice di stringhe, presenta molti vantaggi: prima di tutto è possibile utilizzare direttamente la routine di INSTRING, inoltre non si è limitati a campi di lunghezza fissa, ottenendo così un utilizzo più razionale della memoria del calcolatore. Infine, utilizzando il "trucco" di usare CHR$ 13 (il codice di ENTER) come separatore tra i nominativi si ha la possibilità di stampare in maniera ordinata la rubrica con la semplice istruzione di PRINT a$.

Tenendo presente queste cose è abbastanza semplice modificare il programma fino a trasformarlo in un piccolo archivio, ad esempio prevedendo più campi separati e la possibilità di ordinare i nominativi in base al contenuto di un dato campo.

Note: così com'è il programma va bene sia per lo Spectrum 16 che 48 K, chi ha la versione espansa può però modificare la linea 910, per avere a disposizione più spazio per i dati, in questo modo:

910 CLEAR 64999 LET ins = 65000

Se il programma viene fermato con un BREAK bisogna ripartire con un GOTO 1, evitando assolutamente il RUN, che cancella le variabili.

On Error

Ecco una routine in linguaggio macchina, per la gioia di tutti gli Spectrum-programmatori, principianti od esperti che siano: si tratta della simulazione dell'istruzione ON ERROR, notoriamente assente sui computer Sinclair.

La routine è lunga 200 byte e non è rilocabile, tuttavia il programma di caricamento esadecimale accluso (listato 3) permette di posizionarla a qualunque indirizzo, modificando automaticamente i byte necessari, se non sono stati commessi errori di digitazione il programma termina indicando gli indirizzi delle tre routine ON ERROR GOTO, ON ERROR GOSUB e RESTORE.

Vediamo come si utilizzano questi nuovi comandi

On Error Goto

La sintassi di questo comando è la seguente

POKE *err*,*linea* X USR *add*

dove *add* è l'indirizzo della routine, *err* è un numero tra 0 e 27 che esprime il tipo di errore (vedi il manuale dello Spectrum, appendice B) e *line* è il numero di riga al quale si vuole far saltare il programma quando si trova in presenza dell'errore in questione

Così, ad esempio

POKE 14,100 X USR add

comanda al computer di saltare alla linea 100 in presenza di un errore del tipo "E Out of data"

Per disabilitare un ON ERROR precedente è sufficiente specificare 0 come numero di riga, quindi

POKE 14,0 X USR add

disabilita il comando dell'esempio precedente e fa sì che il computer in seguito si comporti normalmente (almeno nel caso dell'errore di tipo E).

Un'ulteriore possibilità è quella di comandare al computer di "ignorare" semplicemente un errore, passando all'istruzione successiva. Ciò si ottiene specificando un numero di linea uguale a 65535. Ad esempio

POKE 21, 65535 X USR add

comanda al computer di ignorare gli errori del tipo "L BREAK into program" (e quindi costituisce in pratica una sorta di protezione, dal momento che impedisce di fermare un programma premendo il tasto di BREAK)

ON ERROR GOSUB

La sintassi è identica

POKE err,line X USR add

(ovviamente *add* è l'indirizzo della routine ON ERROR GOSUB ed è diverso dal precedente). In questo caso l'errore provoca l'esecuzione di una subroutine, e quando viene incontrato un RETURN il controllo torna all'istruzione successiva a quella che ha causato l'errore

Anche in questo caso è possibile usare 0 o 65535 come numeri di riga (rispettivamente per disabilitare un ON ERROR precedente o per ignorare completamente un errore), ma gli effetti sono identici a quelli specificati per l'istruzione ON ERROR GOTO.

Restore

Abbiamo visto che è possibile ottenere un comportamento diverso per ciascuno dei 28 tipi di errore riconosciuti dal Basic Sinclair (compreso "l'errore" 0, OK), ciò è

Listato 2

```
  1 REM ****************************
  2 REM *                          *
  3 REM *   RUBRICR TELEFONICR     *
  4 REM *                          *
  5 REM ****************************
  6 PEM
 15 DEF FN i(a$,b$,n)=LEN (a$+(
b$ AND n*USR ins))
 20 CLS  PRINT "     RUBRICA TELE
FONICA  ",n;" NOMI
 30 PRINT "Inserisci Ricerca Li
sta Cancella
 40 PRINT #1,'            0 per TE
RMINARE   "
 50 IF INKEY$="I" THEN GO TO 10
0
 60 IF INKEY$="R" THEN GO TO 20
0
 70 IF INKEY$="L" THEN GO TO 30
0
 80 IF INKEY$="C" THEN GO TO 40
0
 85 IF INKEY$="0" THEN GO TO 50
0
 90 GO TO 50
100 REM inserimento
110 PRINT AT 2,0, FLASH 1,"Inse
risci"  BEEP .2,30  PRINT AT 4,0
120 INPUT "NOME e NUMERO TELEFO
NICO"', LINE b$
130 IF b$="" THEN GO TO 20
140 LET a$=a$+b$+CHR$ 13  LET n
=n+1
150 PRINT b$
160 GO TO 120
200 REM    ricerca
210 PRINT AT 2,10, FLASH 1,"Ric
erca". BEEP .2,30  PRINT AT 4,0
220 PRINT : INPUT "Chiave per l
a ricerca"' LINE b$
230 IF b$="" THEN GO TO 20
240 LET find=0
250 LET k=FN i(a$,b$,find+1): I
F NOT k THEN GO TO 280
260 LET k=k-1  IF a$(k)<>CHR$ 1
3 THEN GO TO 260
265 LET find=FN i(a$,CHR$ 13,k+
1)
270 PRINT a$(k+1 TO find),   GO
TO 250
280 IF NOT find THEN PRINT INVE
RSE 1,b$, INVERSE 0," NON E' IN
RUBRICR"'
290 GO TO 220
300 REM    Lista
310 PRINT AT 2,18, FLASH 1,"Lis
ta". BEEP .2,30  PRINT AT 4,0
320 INPUT "". PRINT a$
330 PRINT #1,TAB 9, INVERSE 1,"
PREMI UN TASTO"
340 PAUSE 0  INPUT ""  GO TO 20
400 REM    cancella
410 PRINT AT 2,24, FLASH 1,"Can
cella"  BEEP .2,30  PRINT AT 4,0
420 PRINT   INPUT "Nominativo d
a cancellare"' LINE b$
430 IF b$="" THEN GO TO 20
440 LET find=0
450 LET k=FN i(a$,b$,find+1)  I
F NOT k THEN GO TO 430
455 LET k=k-1  IF a$(k)<>CHR$ 1
3 THEN GO TO 455
460 LET find=FN i(a$,CHR$ 13,k+
1)  PRINT a$(k+1 TO find),
465 PRINT #1,TAB 7, FLASH 1,"CO
NFERMI ?", FLASH 0,"    (S/N)"
470 IF INKEY$<>"S" AND INKEY$<>
"N" THEN GO TO 470
475 INPUT ""
480 IF INKEY$="S" THEN LET n=n-
1  LET a$=a$( TO k)+a$(find+1 TO
) . LET find=k  PRINT "CANCELLAT
O",  BEEP .4,40
485 PRINT   GO TO 450
490 IF NOT find THEN PRINT INVE
RSE 1,b$, INVERSE 0," NON E' IN
RUBRICR"'
495 GO TO 420
500 REM   save
510 BEEP .1,30  PRINT #1,AT 1,1
5, FLASH 1,"TERMINARE"
520 PRINT ''''"      VUOI SALVARE
IL PROGRAMMA"''"          E I DATI
?    (S/N)"
530 IF INKEY$<>"S" AND INKEY$<>
"N" THEN GO TO 530
540 IF INKEY$="S" THEN INPUT ""
  SAVE "RUBRICA" LINE 950  SAVE
"instring"CODE ins,148
550 CLS   STOP
900 REM   inizializzazione
910 CLEAR 31999  LET ins=32000
920 LET a$=CHR$ 13  LET n=0
950 POKE 23658,8  POKE 23609,30
960 LOAD ""CODE ins
970 GO TO 20
```

Listato 3

```
   1 REM ************************
   2 REM *                      *
   3 REM *      ON ERROR        *
   4 REM *     MEX LORDER       *
   5 REM *                      *
   6 REM ************************
   7 REM
  15 OEF FN h(a$)=CODE a$-48-7*(
CODE a$>57)
  20 CLEAR 30000: INPUT "INOIRIZ
ZO = ",add: LET a=add
  25 FOR k=100 TO 180 STEP 10: R
ERD sum: RERO a$
  30 FOR j=1 TO LEN a$ STEP 2: L
ET byte=FN h(a$(j))*16+FN h(a$(j
+1))
  35 POKE a,byte: LET a=a+1: LET
 sum=sum-byte
  40 NEXT j
  45 IF sum<>0 THEN PRINT "ERROR
E IN RIGR ",k: STOP
  50 NEXT k
  55 REM azzeramento tabella
  60 FOR a=a TO a+55: POKE a,0:
NEXT a
  65 REM rilocazione indirizzi
  70 FOR k=1 TO 5: REAO a: REAO
b
  75 POKE add+a,b-256*INT (b/256
): POKE add+a+1,INT (b/256)
  80 NEXT k
  85 PRINT INVERSE 1;"ON ERROR G
OTO"; INVERSE 0,TAB 6,"POKE err,
line*USR ",add
  90 PRINT '' INVERSE 1,"ON ERRO
R GOSUB", INVERSE 0,TAB 6,"POKE
err,line*USR ",add+3
  95 PRINT '' INVERSE 1,"RESTORE
"; INVERSE 0,TAB 6,"RANDOMIZE US
R ",add+123
 100 DATA 1995,"AF18023E01F5C099
1EF1A72802C8F8C6"
 110 DATA 2018,"CD941EC1FE1C02F9
24876F260011009C"
 120 DATA 1744,"19712370ED76305C
333321719CE5C376"
 130 DATA 1339,"18FD7E003C876F26
0011009C194E2346"
 140 DATA 2126,"78CBBFB12835FD36
00FF11719CD578A1"
 150 DATA 2345,"FEFFCA7618CB78CB
B8ED43425CFD360A"
 160 DATA 1961,"01CA7618FD660024
E333ED58455CD5E5"
 170 DATA 1567,"ED73305C11761BD5
C3021F21009C0638"
 180 DATA 1322,"36002310FB2A305C
110313732372E8E9"
 200 DATA 30,add+144,43,add+49,5
8,add+144,75,add+49,124,add+144
```

permesso dal fatto che la routine incorpora una tabella di 56 byte (due per ogni tipo di errore) che contiene i numeri di linea ai quali saltare (oppure 0 se l'errore deve essere trattato normalmente).

L'istruzione RESTORE serve proprio a reinizializzare tale tabella, azzerando i 56 byte e quindi annullando tutti gli ON ERROR precedenti.

La forma di RESTORE è

RANDOMIZE USR add

È importante notare che tale reinizializzazione viene comunque effettuata ogni volta che il programma si arresta per effetto di un errore per il quale non è stato incontrato un ON ERROR precedente.

Note

Il parametro *err* deve essere compreso tra 0 e 27, in caso contrario viene generato un errore del tipo "B Integer out of range" (ma attenzione: l'elaborazione si blocca solo se non esiste un ON ERROR corrispondente).

È decisamente sconsigliabile l'uso di ON ERROR applicato all'errore "C Nonsense in Basic": si perde il contenuto della variabile di sistema CH ADD e spesso lo Spectrum va in crash.

```
org 40000
equ 7833   STK-TO-BC
equ 7828   STK-TO-A
equ 23613  ERR-SP
equ 7030   STMT-RET
equ 23618  NEWPPC
equ 9468   ERROR-B
equ 23621  PPC
equ 7938   TEST-ROOM
equ 4867   ERRGEST
ON-ERROR-GOTO
40000 AF           xor a
40001 18 02        jr ON0
ON-ERROR-GOSUB
40003 3E 01        ld a,1
ON0
40005 F5           push af
[illegible]
40006 CD 99 1E     call STK-TO-BC
[illegible]
ON1
40009 F1           pop af
40010 A7           and a
40011 28 02        jr z,ON2
40013 CB F8        set 7,b
ON2
40015 C5           push bc
[illegible]
40016 CD 94 1E     call STK-TO-A
40019 C1           pop bc
40020 FE 1C        cp 28
40022 D2 F9 24     jp nc,ERROR-B
[illegible]
40025 87           add a,a
40026 6F           ld l,a
40027 26 00        ld h,0
40029 11 D0 9C     ld de,TABELLA
40032 19           add hl,de
40033 71           ld (hl),c
40034 23           inc hl
40035 70           ld (hl),b
[illegible]
40036 ED 7B 3D 5C  ld sp,(ERR-SP)
40040 33           inc sp
40041 33           inc sp
40042 21 71 9C     ld hl,ERROR
40045 E5           push hl
40046 C3 76 1B     jp STMT-RET
[illegible]
ERROR
[illegible]
40049 FD 7E 00     ld a,(iy+0)
40052 3C           inc a
40053 87           add a,a
40054 6F           ld l,a
40055 26 00        ld h,0
40057 11 D0 9C     ld de,TABELLA
40060 19           add hl,de
40061 4E           ld c,(hl)
40062 23           inc hl
40063 46           ld b,(hl)
[illegible]
40064 78           ld a,b
40065 CB BF        res 7,a
40067 B1           or c
40068 28 35        jr z,RESTORE
[illegible]
40070 FD 36 00 FF  ld (iy+0),255
40074 11 71 9C     ld de,ERROR
40077 D5           push de
[illegible]
40078 78           ld a,b
40079 A1           and c
40080 FE FF        cp 255
40082 CA 76 1B     jp z,STMT-RET
[illegible]
40085 CB 78        bit 7,b
40087 CB B8        res 7,b
40089 ED 43 42 5C  ld (NEWPPC),bc
40093 FD 36 0A 01  ld (iy+10),1
40097 CA 76 1B     jp z,STMT-RET
[illegible]
40100 FD 66 0D     ld h,(iy+13)
40103 24           inc h
40104 E3           ex (sp),hl
40105 33           inc sp
40106 ED 5B 45 5C  ld de,(PPC)
40110 D5           push de
40111 E5           push hl
40112 ED 73 3D 5C  ld (ERR-SP),sp
[illegible]
40116 11 76 1B     ld de,STMT-RET
40119 D5           push de
40120 C3 02 1F     jp TEST-ROOM
RESTORE
[illegible]
40123 21 D0 9C     ld hl,TABELLA
40126 06 38        ld b,56
ON4
40128 36 00        ld (hl),0
40130 23           inc hl
40131 10 FB        djnz ON4
[illegible]
40133 2A 3D 5C     ld hl,(ERR-SP)
40136 11 03 13     ld de,ERRGEST
40139 73           ld (hl),e
40140 23           inc hl
40141 72           ld (hl),d
40142 EB           ex de,hl
40143 E9           jp (hl)
[illegible]
TABELLA
dataw 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Disassemblato della routine On Error

Motomuro

di Roberto Allanda - Bergamo

Motomuro è un gioco decisamente divertente, che giro sullo Spectrum 48 K. Nonostante il programma, tra Basic e linguaggio macchina, non sia più lungo di 3 Kbyte, non è possibile adattarlo allo versione con meno Ram a causa del posizionamento di uno dei blocchi all'indirizzo 46848.

Evidentemente l'insensibile autore non ha pensato alle esigenze dei più squattrinati

possessori di Spectrum che per risparmiare si sono comprati il 16 K.

Motomuro è la versione per Spectrum dell'omonimo programma presentato da Valter Di Dio su MC n. 24.

Si gioca in due, a ciascuno dei giocatori lo ZX affida una futuristica moto, dalle strane proprietà. Innanzitutto queste moto non sono dotate di freni, inoltre lasciano una scia radioattiva sopra la quale non si può passare, pena la distruzione della moto su cui si viaggia.

Come si sa, nel mondo dei video game non valgono le leggi della fisica terrestre, per cui questi strani mezzi hanno la capacità di curvare istantaneamente ad angolo retto, senza la minima sbandata. L'obiettivo da raggiungere e ovviamente quello di costringere l'avversario a schiantarsi contro un muro o contro una scia (magari contro quella che lui stesso ha lasciato...)

Quando si è saltati per aria una prima volta lo Spectrum, bontà sua, regala una seconda moto, ma non ci sarà verso di farsene dare una terza. Non esiste una strategia valida con ogni avversario, per vincere bisogna semplicemente avere buoni riflessi e saper pensare velocemente.

Ecco i tasti con i quali si comandono le moto (tra parentesi quelli del giocatore di destra).

ALTO A (J)
BASSO S (K)
DESTRA F (ENTER)
SINISTRA D (L)
STOP Space

La velocità alla quale viaggiano le due moto puo variare tra 1 e 255 (più alto è il numero più si va veloci). Chi riesce a giocare con velocità sopra 200 può tranquillamente concorrere ad un posto di astronauta alla NASA, le persone normali di solito faticano già a 50.

Il programma

La parte principale di Motomuro è scritta in linguaggio macchina, ad un programma Basic sono affidate le funzioni secondarie, come il disegno del campo di gioco e il conto dei punti.

Si può tranquillamente personalizzare il tutto semplicemente disegnando nuovi ostacoli, che il programma riconoscerebbe come tali.

Ecco l'analisi del listato.

1-3 ridefinisce gli UDG "a", "b" e "c" secondo i DATA in coda al programma.

10 azzera i punteggi.

11-12 chiede la velocità delle moto e se viene premuto solamente Enter conserva la velocità precedente.

20-30 inizializza le variabili.

31-43 prepara lo schermo. In questo punto può essere inserita una routine che modifichi il campo di gioco e/o inserisca degli ostacoli.

60 serive i punteggi.

80 chiama il L/M, al ritorno la variabile l conterrà il valore del registro BC.

90-91 se è stato premuto il BREAK il programma si ferma, se l vale 1 va alla linea 100, se vale 2 va alla 200.

100 e 110 salta alla subroutine che disegna le esplosioni.

1000-1002 DATA per gli UDG.

8000-8005 subroutine per l'esplosione ed i suoni corrispondenti.

9000 fine.

Modifiche possibili

Come già accennato, si può modificare il campo di gioco e aggiungere degli ostacoli semplicemente disegnandoli. Ad esempio si potrebbe dividere il campo a metà e vedere quale dei due concorrenti si distrugge per primo.

Listato 4

```
  1 REM ***********************
  2 REM *                     *
  3 REM *  GENERRTORE OELLR   *
  4 REM *      Y TROLE        *
  5 REM *                     *
  6 REM ***********************
  7 REM
  9 CLEAR 31999
 10 LET START=46848
 20 FOR S=1 TO 3
 30 FOR A=0 TO 224 STEP 32
 40 FOR B=0 TO 14 STEP 2
 50 POKE START+B,A
 60 NEXT B
 70 LET START=START+16
 80 NEXT A
 90 NEXT S
100 LET STRAT=46848
110 FOR M=64 TO 88 STEP 8
120 FOR S=1 TO 8
130 FOR A=1 TO 18 STEP 2
140 POKE START+A,M+INT (A/2)
150 NEXT A
160 LET START=START+18
170 NEXT S
180 NEXT M
```

Listato 5

```
   1 REM ************************
   2 REM *                      *
   3 REM *       MOTOMURO       *
   4 REM *                      *
   5 REM ************************
   6 REM
   8 CLEAR 31999  LOAD ""CODE
LOAD ""CODE
   9 RESTORE 1000  FOR a=0 TO 23
 RERD q  POKE USR "a"+a,q. NEXT
 a
  10 POKE 46000,0  POKE 46001,0
  11 INPUT "velocita'? (1/255)="
,v$  IF v$="" THEN GO TO 20
  12 LET v=VRL v$  POKE 46900,IN
T (255/v)
  20 RESTORE : FOR a=1 TO 6  RER
D b,c: POKE b,c  NEXT a
  30 DATR 46100,1,46200,23,46300
,16,46400,240,46301,40,46401,140
  31 PAPER 7  INK 1  BORDER 7  C
LS
  40 PRINT AT 1,0,"
  41 FOR a=2 TO 20  PRINT RT a,0
,"
    NEXT a
  42 PRINT RT 21,0,"
  43 PRINT RT 3,0,"
  60 PRINT RT 2,5,"SCORE1=",PEEK
 46000,"          SCORE2=",PEEK 460
01
  80 LET l=USR 32000
  90 IF l=0 THEN STOP
  91 IF l=2 THEN GO TO 110
 100 LET x=PEEK 46300  LET y=PEE
K 46301. GO SUB 8000  IF PEEK 46
001<3 THEN GO TO 20
 101 CLS  PRINT "Hai vinto tu,n
umero 2"  GO TO 300
 110 LET x=PEEK 46400  LET y=PEE
K 46401  GO SUB 8000  IF PEEK 46
000<3 THEN GO TO 20
 111 CLS  PRINT "Hai vinto tu,n
umero 1"  GO TO 300
 300 PRINT RT 10,3,"Un'altra par
tita?"  PAUSE 0
 301 IF INKEY$="s" THEN GO TO 1
 302 STOP
1000 DRTA BIN 0,BIN 01000010,BIN
 00010000,BIN 00111000,BIN 10110
000,BIN 00010100,BIN 00100000,0
1001 DATA BIN 01000000,BIN 11010
100,BIN 00000010,BIN 01010010,BI
N 00011000,BIN 10000001,BIN 1000
0010,BIN 01101000
1002 DATA BIN 00010101,BIN 01000
000,BIN 10000000,BIN 10000001,BI
N 00000001,BIN 01000001,BIN 0000
1000,4
8000 LET x=INT (x/8)  LET y=INT
(y/8)
8001 PRINT AT y,x,"A". BEEP .2,-
55
8002 PRINT AT y,x,"B". BEEP .2,-
45
8003 PRINT RT y,x,"C". BEEP .3,-
50
8004 PRINT RT y,x," "
8005 RETURN
9000 STOP
```

Listato 6

indirizzo	codici					
32000	62	253	219	254	230	31
32006	254	31	40	3	50	20
32012	180	58	20	180	230	30
32018	254	30	32	10	58	221
32024	180	61	50	221	180	195
32030	77	125	58	20	180	230
32036	29	254	29	32	10	58
32042	221	180	60	50	221	180
32048	195	77	125	58	20	180
32054	230	27	254	27	32	10
32060	58	220	180	61	50	220
32066	180	195	77	125	58	220
32072	180	60	50	220	180	58
32078	220	180	230	7	71	62
32084	7	144	50	164	181	58
32090	220	180	203	63	203	63
32096	203	63	50	108	182	58
32102	221	180	111	62	0	103
32108	41	124	198	183	103	94
32114	44	86	213	225	58	108
32120	182	133	111	126	95	58
32126	164	181	71	123	4	7
32132	15	16	253	230	1	254
32138	1	202	95	126	58	164
32144	181	71	95	126	4	7
32150	15	16	253	203	199	67
32156	4	15	7	16	253	119
32162	62	191	219	254	230	31
32168	254	31	40	3	50	120
32174	180	58	120	180	230	23
32180	254	23	32	10	58	65
32186	181	61	50	65	181	195
32192	239	125	58	120	180	230
32198	27	254	27	32	10	58
32204	65	181	60	50	65	181
32210	195	239	125	58	120	180
32216	230	29	254	29	32	10
32222	58	64	181	61	50	64
32228	181	195	239	125	58	64
32234	181	50	50	64	181	58
32240	64	181	230	7	71	62
32246	7	144	50	164	181	58
32252	64	181	203	63	203	63
32258	203	63	50	108	182	58
32264	65	181	111	62	0	103
32270	41	124	198	183	103	94
32276	44	86	213	225	58	108
32282	182	133	111	126	95	68
32288	164	181	71	123	4	7
32294	15	16	253	230	1	254
32300	1	40	60	58	164	181
32306	71	95	126	4	7	15
32312	16	253	203	199	67	4
32318	67	4	16	7	16	253
32324	119	62	127	219	254	230
32330	1	254	1	32	15	58
32336	52	183	71	197	6	0
32342	18	254	193	16	248	195
32348	0	125	201	58	177	179
32354	60	50	177	179	6	0
32360	14	1	201	58	176	179
32366	60	50	176	179	6	0
32372	14	2	201	0	0	0

Linguaggio macchina.

Listato 7

```
   1 REM ************************
   2 REM *                      *
   3 REM *    MOVEA GRAFICO     *
   4 REM *                      *
   5 REM ************************
   6 REM
  10 DATA 62,127,50,27,189,50,29
,189,82,87,50,28,189,50,30,9455
  20 DATA 189,175,50,24,189,62,1
75,50,25,189,50,41,189,50,40,106
41
  30 DATA 189,205,246,188,205,11
8,186,17,0,0,237,83,141,186,237,
16855
  40 DATA 83,161,186,205,246,188
,205,118,186,58,27,189,71,58,29,
13509
  50 DATA 189,184,48,7,50,27,189
,120,50,29,189,68,28,189,71,1101
4
  60 DATA 58,30,189,184,56,7,50,
28,189,120,50,30,189,58,30,9847
  70 DATA 189,71,58,28,189,144,6
0,50,36,189,58,29,189,230,7,1222
8
  80 DATA 60,71,175,55,31,16,252
,50,31,189,58,27,189,71,58,10814
  90 DATA 29,189,144,50,24,189,6
8,30,189,71,58,28,189,144,71,118
62
 100 DATA 62,175,144,50,25,189,1
75,50,41,189,17,28,189,237,83,13
730
 110 DATA 141,186,17,27,189,237,
83,161,186,205,226,187,175,50,40
,16937
 120 DATA 189,205,246,188,205,11
8,186,201,205,60,187,205,205,186
,203,22008
 130 DATA 111,40,6,205,201,187,1
95,118,186,203,71,192,33,30,189,
15938
 140 DATA 17,28,189,203,79,40,3,
62,235,52,203,87,40,3,63,9793
 150 DATA 235,53,33,29,189,17,27
,189,203,95,40,3,52,235,52,11303
 160 DATA 203,103,40,3,53,235,53
,79,58,41,189,183,32,190,121,133
17
 170 DATA 203,119,40,6,205,12,18
8,195,118,186,203,127,40,3,205,1
4874
 180 DATA 19,188,195,118,186,22,
0,1,254,251,237,120,203,79,32,15
590
 190 DATA 13,58,30,189,33,25,189
,190,40,4,203,202,24,17,1,9755
 200 DATA 254,254,237,120,203,87
,32,8,58,30,189,183,40,2,203,124
68
 210 DATA 210,1,254,253,237,120,
203,79,32,2,203,234,203,87,32,15
630
 220 DATA 11,58,29,189,254,255,4
0,4,203,218,24,17,237,120,203,16
383
 230 DATA 71,32,11,58,29,189,33,
24,189,190,40,2,203,226,1,11985
 240 DATA 254,191,237,120,203,71
,32,2,203,194,203,87,32,4,203,14
069
 250 DATA 242,24,6,203,95,32,2,2
03,250,122,201,6,2,197,58,12832
 260 DATA 27,189,79,58,28,189,71
,205,190,187,205,229,34,14,0,135
06
 270 DATA 58,27,189,71,58,29,189
,205,129,187,14,1,58,28,189,1156
8
 280 DATA 71,58,30,189,205,129,1
87,14,0,58,29,189,71,58,27,9560
 290 DATA 189,205,129,187,14,1,5
8,30,189,71,58,28,189,205,129,13
103
 300 DATA 187,193,16,190,201,13,
40,22,144,48,3,237,68,55,245,129
02
 310 DATA 205,40,45,241,48,3,239
,27,58,239,160,56,195,177,187,16
675
```

Si può anche modificare il punteggio di vittoria cambiando i 3 alle linee 100 e 110. Al campione poi si potrebbe dare un handicap, assegnandogli un tetto superiore allo sfidante.

Caricamento in memoria

Per registrare su nastro Motomuro dovrete eseguire le seguenti operazioni:

1) digitate il listato 4, date RUN e poi NEW

2) Inserite il linguaggio macchina del listato 6 a partire dall'indirizzo 32000, facendo molta attenzione a non commettere errori. Per l'inserimento va bene qualsiasi caricatore decimale.

3) Digitate il listato 5

4) Date il comando diretto SAVE "MOTOMURO": SAVE "L/M1" CODE 32000, 375 SAVE "L/M2" CODE 46848, 385

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 157.

Mover grafico per Melbourne Draw

di Domenico Manildo - Padova

Non so se "mover" sia il nome adatto per il programma che vi presento: lo si potrebbe anche chiamare "copyst" e per certi versi gli si adatterebbe persino il nome di "selective eraser" (maledetta anglofilia! I tre verbi move, copy ed erase significano, rispettivamente, spostare, copiare e cancellare. N d r)

Insomma sono molti gli scopi per cui un tale programma può essere utilizzato, e li illustrerò con un esempio.

Sto preparando la schermata iniziale di un gioco che ho inventato, ed ora che il disegno è pronto voglio aggiungervi il titolo, con dei bei caratteri grafici disposti a semicerchio. Mi accorgo di avere una pagina grafica in cui ho disegnato tutti i caratteri dell'alfabeto, belli grandi, scritti con uno stile molto appariscente. Carico allora in memoria, nella seconda pagina del mio "mover" questa schermata ed aziono il programma. Richiamo la seconda pagina e, muovendo con gli appositi tasti il puntino che lampeggia sullo schermo, mi avvicino alla prima lettera che voglio far apparire nel disegno, la circondo con un rettangolo lampeggiante e la memorizzo. Poi cambio pagina, ritornando al mio disegno: muovo il rettangolo nel posto dove voglio che appaia la lettera, premo un tasto e la lettera compare, senza cancellare il disegno sottostante ma sovrapponendovisi, a mia scelta, in OVER o in modo semplice.

Ripeto l'operazione per tutte le lettere del titolo ed alla fine, con poca fatica, ho composto la scritta.

Fin qui ho utilizzato il programma come "copyst". Mi accorgo però che una lettera è fuori posto, non si inserisce bene nel semicerchio formato dalle altre.

Aziono allora nuovamente il programma, circondo la lettera con il solito rettangolo, la memorizzo e quindi premo il tasto che riversa il disegno in modo OVER: la lettera ovviamente scompare.

A questo punto posiziono il rettangolo nel punto più adatto e vi riverso la lettera, che posso ancora eventualmente cancellare e spostare, fino a che non sia al posto giusto.

Ecco che, in questo caso, ho utilizzato il

```
 320 DATA 245,197,239,160,56,193
,241,144,48,3,237,68,55,245,205,
17378
 330 DATA 40,45,241,48,3,239,27,
56,217,229,217,205,190,187,205,2
0384
 340 DATA 119,36,217,225,217,201
,33,145,92,203,198,62,255,50,144
,17379
 350 DATA 92,201,33,176,216,17,0
,64,1,0,27,197,126,71,26,8683
 360 DATA 119,120,18,35,19,193,1
1,120,177,32,241,201,205,179,188
,17610
 370 DATA 34,34,189,58,28,189,50
,26,189,33,176,189,34,38,189,127
77
 380 DATA 237,75,36,189,197,205,
63,188,205,209,188,205,136,188,2
05,21349
 390 DATA 77,188,193,11,120,177,
32,237,201,62,182,50,239,188,24,
15988
 400 DATA 5,62,174,50,239,186,33
,176,189,34,38,189,58,28,189,135
21
 410 DATA 50,26,189,205,179,188,
237,75,36,189,197,205,63,188,205
,19138
 420 DATA 77,188,205,105,188,205
,209,188,193,11,120,177,32,237,2
01,18665
 430 DATA 33,42,189,54,0,229,209
,19,1,32,0,237,176,201,42,12794
 440 DATA 38,189,17,42,189,237,7
5,34,189,58,40,189,183,245,40,15
197
 450 DATA 1,235,237,176,241,40,1
,235,34,38,189,201,58,27,189,143
62
 460 DATA 230,7,200,71,197,33,42
,189,183,245,58,32,189,60,71,135
98
 470 DATA 241,126,31,245,119,35,
16,248,241,193,16,233,201,237,91
,18834
 480 DATA 32,189,33,42,189,29,25
,58,31,189,166,119,58,27,189,118
25
 490 DATA 230,7,200,71,197,229,5
8,32,189,71,183,245,241,126,23,1
6755
 500 DATA 119,245,43,16,248,241,
225,193,16,235,201,58,28,189,71,
16083
 510 DATA 58,27,189,79,197,205,1
70,34.193,58,29,189,79,229,205,1
6334
 520 DATA 170,34,209,183,237,82,
44,34,32,189,201,58,26,189,71,12
988
 530 DATA 61,50,26,189,58,27,189
,79,205,170,34,58,32,189,71,1214
4
 540 DATA 17,42,189,58,40,189,18
3,40,1,235,26,182,119,35,19,1098
6
 550 DATA 16,249,201,1,32,78,183
,11,120,177,32,250,201,33,0,1239
5
 560 DATA 128,17,0,64,1,0,27,237
,176,201,33,0,64,17,0,7535
 570 DATA 128,1,0,27,237,176,201
,0,0,0,0,0,0,0,0.3986
1000 CLEAR 47560  RESTORE   LET
d=47561
1100 FOR i=1 TO 57
1120 LET check=0
1130 FOR c=1 TO 15
1140 READ a  POKE d,a  LET d=d+1
  LET check=check+a*c
1150 NEXT c
1160 READ cont  IF cont<>check T
HEN GO TO 2000
1170 NEXT i
1180 SAVE "MOVECARIC.". SAVE "MO
VECODE"CODE 47561,850
1190 CLS   PRINT "RIAVVOLGI E VE
RIFICA!"  VERIFY "". VERIFY ""CO
DE   STOP
2000 CLS   PRINT "ERRORE ALLA LI
NEA ",i*10
```

programma sia come "selective eraser" che come "mover"

L'idea di servere questa utility mi e venuta vedendo qualcosa di simile funzionare su di un Commodore 64 dotato di tavoletta grafica. Sul C 64, grazie al modo grafico a media risoluzione, che permette di definire il colore di ogni singolo punto, vengono copiati anche i colori.

Nel mio programma ho dovuto purtroppo rinunciare a questa possibilità perche avrebbe funzionato solo se il disegno fosse stato riversato in punti ben precisi dello schermo

Per quanto non abbia bisogno di alcun supporto esterno per funzionare, questo programma risulta molto più agevole da usare se inserito come opzione in un programma di grafica, come l'ottimo Melbourne Draw Ho pensato quindi di preparare le righe necesarie per modificare il menu principale di quest'ultimo, al fine di poter utilizzare con esso il mover.

Dettagli tecnici

Il programma è scritto interamente in linguaggio macchina ed è lungo 847 byte I codici decimali del programma sono riportati nelle linee DATA dell'apposito caricatore (listato 7), che prevede una verifica automatica dei dati per controllare eventuali errori di battitura

La sequenza delle operazioni da eseguire per inserire il mover nel Melbourne Draw e la seguente.

— scrivere il programma del listato 7 e dare il RUN Se non sono stati commessi errori il programma salva sia se stesso che il linguaggio macchina

— dare il comando diretto RANDOMIZE USR 0

— caricare il Melbourne Draw ed entrare in Basic con l'apposita opzione

— aggiungere al listato di Melbourne Draw le linee del listato 8

— caricare da nastro il codice macchina del mover precedentemente salvato

— dare il comando GOTO 9900 per salvare su nastro il Melbourne Draw modificato

Istruzioni per l'uso

Appena caricato il Melbourne Draw appaiono due nuove opzioni {m} e {2}. Con m si accede al mover, mentre con 2 si carica da nastro la seconda pagina grafica, che viene messa nella parte alta della memoria Ad essa si può accedere solo dal mover, perche in genere contiene disegni o lettere da prelevare e riversare sulla pagina principale

Quando si richiama il mover appare la pagina principale e, al centro, si può vedere un puntino lampeggiante.

Esso può essere spostato senza cancellare il disegno sottostante, così come i rettangoli di cui parleremo fra poco

Il lavoro con il mover si articola in tre fasi

— posizionamento del primo vertice del rettangolino

— posizionamento del secondo vertice del rettangolino, con contemporanea visione della zona così circoscritta

— posizionamento del rettangolino nella zona dove il disegno va riversato e riversamento dello stesso

Ciascuna fase va terminata con la pressione del tasto Enter, prima dell'ingresso nella fase successiva viene eseguito un ciclo di ritardo, per evitare che, tenendo inavvertitamente premuto il tasto per troppo tempo, si passi direttamente dalla prima all'ultima fase

Il movimento viene comandato con i seguenti tasti

W-ALTO
X-BASSO
A-SINISTRA
D-DESTRA

La figura a sinistra è stata ottenuta con il mover a partire dalla celebre schermata di presentazione dell Hobbit Le lettere che formano la scritta MC MICROCOMPUTER sono state prelevate dalla pagina grafica mostrata nella figura di destra che è stata ottenuta con il programma DLAN della Campbell System

Listato 8

```
********************************
*                              *
*     LINEE DA RGGIUNGERE      *
*      A MELBOURNE DRAW        *
*                              *
********************************

  91 PRINT    {m} ) call mover
 155 PRINT '  {2} ) load second
page
 235 IF a$= m  OR a$= M  THEN GO
TO 2000
 236 IF a$= 2  THEN GO TO 2100
2000 RANDOMIZE USR 48384
2010 RANDOMIZE USR 47561  RANDOM
IZE USR 48398
2020 GO TO 30
2100 CLS   GO SUB 300  LOAD n$CO
DE 88472  LET n$= Load'  GO TO 1
000
9900 SAVE "MEL" LINE 10  SAVE "m
elc" CODE 40980,7600
9910 CLS   PRINT  RIAVVOLGI E VE
RIFICA'
9920 VERIFY ""  VERIFY ""CODE
```

Il tasto S, attivo in tutte le fasi, permette di scambiare la prima con la seconda pagina grafica Quando lo si preme, la seconda pagina appare sul video, mentre la prima viene salvata al posto della precedente Fate attenzione che, all'uscita dalla terza fase, tutto quello che si trova sul video viene riversato nella pagina principale, e sarà accessibile da Melbourne Draw Se per errore ci dovesse essere la seconda pagina, basterà richiamare il mover, scambiare le due pagine con S ed uscire dal mover premendo tre volte Enter

Nella terza fase risultano abilitati anche altri due tasti

J-riversa il disegno eseguendo un OR esclusivo con lo schermo Ciò significa che il suo effetto risulta annullato se viene premuto per due volte consecutive, cosa che risulta molto utile in fase di posizionamento del disegno, in quanto permette ripensamenti

Per cancellare la zona circondata è sufficiente premere J appena entrati nella terza fase

K-riversa il disegno eseguendo un OR semplice

E possibile tenere premuto un tasto di riversamento mentre si sposta il rettangolino, ottenendo un effetto scia che può talvolta risultare gradevole

In ogni caso non c'è limite al numero di volte che uno stesso disegno può venire riversato Il ritorno al menu principale si ha solo sempre con Enter

Naturalmente il programma funziona anche al di fuori di Melbourne Draw ed è perciò possibile inserirlo in altri programmi

Per attivarlo si deve usare l'istruzione RANDOMIZE USR 47561, purtroppo non è rilocabile in quanto contiene troppi indirizzi assoluti. La seconda pagina si trova in memoria a partire dalla locazione 55472, e può essere caricata da nastro con un LOAD "" CODE 55472 MC

I COMPUTER MPF PER POTER SCEGLIERE
COMPUTER DIDATTICI MPF (MICROPROFESSOR)
MPF I/ P basato su Z80 8 bit
MPF I/65 basato su 6502 8 bit
MPF I/88 basato su 8088 16 bit
MPF I/68 basato su 68000 32 bit
MPF I
HOME/PERSONAL COMPUTER
MPF II
CPU 6502, 1 Mhz/ROM 16 K con interprete basic apple soft, Monitor Disassembler / Ram 64 K
MPF III
CPU 6502 1 MHz / ROM 24 K (con interprete basic) / RAM 64 K dinamiche piu 2 K statiche per le 88 colonne di testo / TESTO 40 x 24 - 80 x 24
TASTIERA distaccata con 90 tasti multifunzione
MPF II
MPF III
PERSONAL/PROFESSIONAL COMPUTER
MPF PC - MPF PC/XT
CPU 8088 piu 8087 (opzionale)
ROM 8 K espandibili a 48 K su scheda
RAM 128 K espandibili a 256 K su scheda e 640 K esternamente
Disk driva MPF PC 2 disk drive per 720 K
Disk drive MPF PC/XT 1 disk drive per 360 K piu HD da 10 Mb
Interfacce resistenti 1 RS232 + centronic + controller disk drive + scheda colore + 4 slot compatibili disponibili utente
Sistema operativa standard Concurrent CP/M86 con PC MODE
Sistema operativo opzionala MS DOS
MPF PC/XT
MPF PC
DIGITEK COMPUTER
VIA VALLI, 28 - 42011 BAGNOLO IN PIANO (Reggio Emilia)
Tel. (0522) 61623 r.a. - Telex 530156

Scopa - *di Mauro Casartelli - Como*

Nello scorso mese di settembre pubblicammo un programma che permetteva di giocare a briscola contro il computer, oggi vi presentiamo questo programma, a cura dello stesso autore di "Briscola", che vi permetterà di sfidare il calcolatore in una avvincente partita a scopa.

Quando abbiamo osservato il gioco, per la prima volta, il nostro pensiero è subito andato all'analogo programma per il PC IBM che avevamo avuto modo di vedere all'opera nella recente trasmissione televisiva CHIP di RAI 2, e che senz'altro molti di voi ricorderanno. Ebbene, possiamo assicurarvi che il programma che vi presentiamo ha retto benissimo il confronto, anzi si è dimostrato talvolta superiore per alcune scelte azzeccate nella veste grafica.

Anche questa volta, come già nel caso della briscola, assumeremo che siano già note a tutti le regole del gioco, vorremmo soltanto osservare, prima di passare a illustrarvi il programma, quanto incida la fortuna in questo gioco di carte (anche più che nella briscola), tanto che facendo giocare il computer contro se stesso si sono verificati ripetutamente squilibri davvero notevoli nei punteggi parziali e globali.

Le modalità del gioco sono le consuete: mazzo di quaranta carte, quattro carte in tavola all'inizio della partita, distribuzione ogni mano di tre carte a ciascuno dei tre giocatori. La carta calata prende una carta di egual valore o, in assenza di questa, due o più carte di uguale valore complessivo. Si fa "scopa" quando dopo la presa non rimangono carte in tavola, purché ciò non avvenga alla conclusione della partita.

Le carte in tavola sono rappresentate nella zona superiore dello schermo ciascuna sottoindicata da una lettera minuscola di riferimento. Ogni carta che viene calata viene rappresentata sempre nella parte superiore del video in posizione il più centrale possibile. Le carte in mano al computer sono rappresentate, con il disegno del dorso, nella parte inferiore destra del video, quelle in mano al giocatore nella parte inferiore sinistra, ciascuna sottoindicata dal nome del tasto da premere per scartarla.

Il mazziere riceve sempre le carte per secondo (e gioca per secondo), comunque un segnale acustico avverte il giocatore quando è il suo turno. l'ultima carta calata e le eventuali carte oggetto di una presa vengono evidenziate cambiandone il carattere di sfondo. L'utente sceglie cosa giocare premendo il tasto con la cifra di riferimento della carta prescelta, se tale carta non prende o prende univocamente, il programma prosegue automaticamente, altrimenti un suono intermittente ricorda al giocatore di premere i tasti (non shiftati, alpha lock alzato) con la lettera di riferimento delle carte prescelte per la presa. se il totale valori carte prescelte è inferiore al valore di presa, il programma attende altre indicazioni. se tale totale eccede il valore di presa, un segnale sonoro prolungato avverte dell'errore e bisogna ricomunicare tutte le carte prescelte, se tale totale eguaglia il valore di presa, il programma riparte normalmente. La scopa viene confermata con un opportuno segnale acustico. Il totale carte ancora da distribuire viene indicato dall'altezza di una colonnina di caratteri stampata nella zona inferiore centrale dello schermo. Il mazziere varia alternativamente partita dopo partita (il primo mazziere è sempre l'utente). le carte rimaste in tavola a fine partita vengono automaticamente assegnate a chi ha preso per ultimo.

Per esplicitare l'attribuzione dei punti ai contendenti, a fine partita per scope, ori e settebello, primiera, si succedono tre schermate con stampa a sagoma intera delle carte (giocatore: zona destra dello schermo, computer: zona sinistra) considerate nel rispettivo calcolo. Il totale carte prese viene visualizzato in una quarta schermata (in effetti dopo le scope) dall'altezza di due opportune colonne di caratteri, una quinta schermata comunica, per ciascuno dei due contendenti, il risultato finale della partita, il punteggio (somma dei punti fin lì accumulati), il numero delle vittorie. Mentre al termine di ciascuna delle prime quattro schermate il programma si arresta e riparte solo dopo avere premuto un tasto qualunque, al termine di questa quinta schermata si offrono esplicitamente all'utente due possibilità: cominciare una nuova partita, oppure farsi visualizzare il riepilogo punti, cioè quanti punti sono stati globalmente conquistati, nelle partite fin lì giocate, da ciascuno dei due giocatori per ciascuna delle cinque voci (scope, carte, ori, settebello, primiera) che concorrono alla formazione del punteggio. Tutti questi ultimi dati numerici, come pure punteggio e vittorie, sono stampati anche in una particolare forma grafica: per ogni voce viene stampata una linea bicolore (bianca e rossa) di lunghezza totale fissa (20 caratteri), ma nella quale il rapporto tra la lunghezza del tratto bianco e la lunghezza del tratto rosso è uguale al rapporto tra i corrispondenti dati numerici di computer e giocatore (l'unità di misura per calcolare e disegnare le due parti della linea è ¼ di carattere). I tempi di gioco per ogni partita sono: preliminari circa 25 secondi, fase di gioco: circa 3 minuti, calcolo e stampa risultati finali: circa 35 secondi, eventuale stampa e riepilogo punti: circa 9 secondi, totale: 4 minuti abbondanti. Il testo del programma occupa 7332 byte, in esecuzione rimangono liberi circa 3900 byte (usando il TI BASIC).

Convenzioni e meccanismi

1) Le quattro carte in tavola a inizio partita hanno sempre quattro valori tra loro diversi perché gran parte dei meccanismi adottati funziona correttamente, all'attuale grado di sviluppo del programma, solo sotto tale condizione.

2) Ogni carta è rappresentata internamente al programma come codice numerico ottenuto sommando al codice seme (1, 2, 3, 4 per cuori, quadri, fiori, picche) il codice valore (0, 1 .., 8, 9 per asso, due, .., donna, re).

3) Il mazzo, vettore K (0 39), viene mischiato ponendovi dapprima le carte in ordine crescente e scambiando poi la carta contenuta in ciascuna delle quaranta posizioni con una carta scelta a caso nelle prime trentasei posizioni (ciclo di scambio ripetuto due volte, quindi ottanta permutazioni, per avere un buon mescolamento).

4) Il calcolo delle combinazioni di presa realizzate dalle carte in tavola è reso pressoché immediato dall'introduzione dei seguenti vettori:

A$ (0 41), inizializzato con le stringhe di caratteri coi valori delle carte che realizzano le 42 diverse possibili combinazioni di presa (es. A$(3)="01" indica come presa valida asso + due).

F (0 41), contiene il numero delle carte presenti in tavola appartenenti a ciascuna delle 42 combinazioni, es. F(3)=2 indica che in tavola vi sono due carte della combinazione A$(3)="01", inoltre essendo LEN(A$(3))=2 e non essendoci mai in tavola due carte di ugual valore (vedi punto 1), ciò significa che asso + due è una presa realizzabile in quella fase di gioco.

B$ (0 9), inizializzato con stringhe di caratteri atte a fornire una corrispondenza tra valore carta e combinazioni di presa alle quali tale valore appartiene (es. B$(8)="@1" indica che la carta di valore 8 (donna) è presente nelle combinazioni identificate in A$(0 41) dagli indici (ASC("@")-40)=24 e (ASC("1")-40)=33, infatti A$(24)="8" e A$(33)="08"), l'introduzione di questo vettore rende più veloce l'aggiornamento di F (0 41) necessario ogni volta che una carta viene tolta dal tavolo o aggiuntavi definitivamente.

V1(0 9), V2(0..9), inizializzati con valo-

ri tali per cui, data una carta di valore n, le combinazioni di presa per essa possibili stanno tra A$(V1(n)) eA$(V2(n)) compresi, es n=2 (la carta è un tre), V1 (2)=2 e V2(2)=3, quindi le combinazioni di presa potenziali giocando un tre sono A$(2)="2" e A$(3)="01" cioe tre e asso+due, per sapere se tali prese sono o meno realizzabili bisogna poi guardare i valori di F(2) e F(3) rispettivamente

5) La stampa delle carte sullo schermo avviene in posizioni (riga e colonna) predefinite e per queste tre situazioni, carte in mano ai giocatori, carte in tavola, carte considerate per i due giocatori per conteggiare scope, ori e settebello, primiera Tutti questi valori di riga (rispettivamente 6, 11, 10+10) e di colonna (altri 37 valori) sono prememorizzati consecutivamente in un'unica stringa nella variabile X$, ciascuno come CHR$ (65+(riga o colonna)). Di conseguenza la subroutine di stampa delle carte non riceve come parametri riga e colonna, ma calcola tali valori da X$ servendosi dei valori delle variabili ST (e ST+37, posizioni di partenza in X$ per la determinazione delle coordinate di stampa), dipendente dalla situazione, ed N, indice di posizione di stampa in ogni situazione. Questo meccanismo permette un buon risparmio di memoria essendo tra l'altro l'indice N comunque già utilizzato dal programma per altri scopi, inoltre quando si esegue un'operazione ciclica, la posizione di stampa è vincolata all'indice N del ciclo, ma i valori di riga e colonna non ne risultano per esempio funzione crescente

6) Il computer dispone di una strategia di gioco abbastanza semplice tendente a fargli prendere le carte più interessanti ai fini del risultato finale, senza peraltro rischiare troppo di subire scopa (ed è soprattutto a tal fine che vengono ricordate le carte già andate) In alternativa il computer dispone pure di una strategia più rude che lo porta a giocare in maniera più spregiudicata, ma meno prevedibile In base al risultato finale dell'ultima partita giocata ed in base ai punteggi conseguiti usando le due strategie, il computer sceglie di giocare nel modo tendenzialmente più affidabile

NOTA modificando la linea 630 come segue 630 X=124 il computer vi mostra le sue carte

7) Il programma non usa funzioni predefinite (DEF.) perché ne rallenterebbero l'esecuzione, nella dichiarazione DIM non compaiono i vettori di 11 elementi, già implicitamente definiti, vi compaiono invece tutti quelli di meno di 11 elementi per evitare che il programma riservi per essi memoria oltre il necessario, non viene fatta giocare automaticamente l'ultima carta rimasta in mano all'umano per evitare nel caso ci si debba assentare di rischiare di perdere la visione di due turni di gioco del computer, se uno dei due giocatori realizza in una stessa partita più di dieci scope si genera errore nella stampa finale delle stesse (premesso che non è mai capitato, potrebbe però succedere a causa di un divario di forze tale che sarebbe comunque meglio smettere di giocare!!) Vale la pena di ricordare a chi volesse copiare il testo usando l'extended Basic, che nelle istruzioni print i due punti consecutivi (::) vanno sempre spaziati tra loro (: :)

Variabili principali

(n, nl=indici vettori, v=valore carta (da 1 a 10))

A$(0 41), B$(0 9), F(0 41), V1(0 9), V2(0 9) vedi punto 4)

X$,ST vedi punto 5)

CC possibilità di presa =0 nessuna possibilità, =1 presa univoca, =2 diverse combinazioni di presa realizzabili

CG(0 2): coefficiente di giocabilità di ciascuna delle carte in mano al computer

CN coefficiente tanto maggiore quanto più conveniente è ritenuta dal computer la presa realizzabile esaminata

DF(0 1) scarto totale nel punteggio tra computer e umano (n=1 solo scarto partite giocate con la strategia evoluta, n=0 idem per la strategia rude)

DS(0 2). per ciascuna delle carte in mano al computer contiene il valore dell'indice DD o D tale che A$ (DD o D) è la combinazione di presa da realizzare se la carta giocata è prendente

E$ stringa di caratteri corrispondenti ai valori (v-1) delle carte da prendere (es E$="01" prendere asso+due)

G turno di gioco, =0 gioca il computer =1 gioca l'umano

H(0 9) posizione in tavola della carta di valore v=n+ 1 (se in tavola non c'è la carta di valore v=n+1, allora H(n) = -1)

I(0 10) codice della carta presente in ciascuna delle undici predefinite posizioni di stampa delle carte in tavola (se non c'e carta in tavola nella posizione n allora I(n)=0)

J(0 1,0 39) codici delle carte prese (n=0 dal computer, n=1 dall'umano)

K(0. 39) mazzo, le carte in tavola a inizio partita sono in K (36) e seguenti

P(0 1) punti conquistati nell'ultima partita giocata (n=0 dal computer; n=1 conquistati dall'umano)

PU(0 9) punteggi di primiera per i dieci valori delle carte (n=v-1).

PV somma dei valori v delle carte in tavola

Q posizione in tavola dell'ultima carta stampatavi

QT(0 10) per ciascun valore quante carte vi sono ancora complessivamente nel mazzo ed in mano all'avversario umano (n=v) (e fondamentale per il corretto funzionamento della strategia del computer che il valore di QT(0), sempre uguale a zero, non venga mai modificato)

S(0 1) numero di carte prese (n=0 dal computer, n=1 dall'umano)

SG strategia del computer, =1 evoluta, =0 rude

U numero dei turni di gioco già completati e dunque legato al numero di carte già giocate

V(0 6,0. 1): tutti i dati numerici globali riportabili a fine partita: n=0 punti totali, n=1 vittorie, n=2 scope, n=3 punti conquistati con le carte, n=4 idem per gli ori, n=5 idem per il settebello, n=6 idem per la primiera (n1=0 computer, n1=1 umano)

W attributore di presa =0 presa del computer, =1 presa dell'umano

X codice ASCII del carattere di stampa sfondo carte (X=32 per cancellare la carta).

Z(0 .3,0 1): per ciascuno dei quattro semi il codice della carta con il maggiore punteggio di primiera (n1=0 computer, n1=1 umano)

Variabili utilizzate in contesti diversi

A variabile locale nel pre e post partita, durante la fase di gioco il programma fa riferimento al suo valore per stampare il carattere di sfondo carta evidenziato per tutte le carte giocate dai giocatori ma non per quelle quattro distribuite in tavola ad inizio partita

C variabile locale, alla fine di ogni partita ne indica il vincitore =0 vittoria computer, = 5 pareggio, =1 vittoria umano.

L(0 5) carte in mano all'umano (n=0, 1, 2) e al computer (n=3, 4, 5), quando la carta in posizione n è già stata giocata L(n)=0 Per risparmiare memoria, questo vettore è anche usato a fine partita per memorizzarvi i punteggi di primiera del computer (in L(0)) e dell'umano (in L(1))

N per lo più indice dei vettori collegati al trattamento delle carte, in quanto tale è usato nella determinazione della posizione di stampa delle carte stesse (vedi punti 4) e 5))

R variabile locale, rappresenta in tutte le procedure il codice della carta trattata

R$(0 1) contiene, memorizzati in tempi successivi, i codici convertiti alfanumericamente, ora delle carte che hanno fatto scopa, ora delle carte di quadri, ora delle carte di primiera (n=0 computer, n=1 umano)

M$, T$ usate invece di vettori nelle procedure di stampa dei dati riassuntivi per ricavarne opportuni segmenti di stringa

B, D, E, LA, M, T, C$, D$ variabili locali in molte procedure (l'ultimo valore assegnato a D$ viene usato in tutto il programma per definire il carattere evidenziatore dello sfondo carta)

Analisi del listato

10-220 inizializzazioni vettori e variabili e ridefinizione caratteri per un corretto svolgimento del gioco. Essendo tali operazioni necessarie una tantum vengono eseguite solo all'inizio della prima partita giocata dopo il RUN Se si interrompe l'esecuzione del programma ne consegue che alcuni colori e caratteri perdono (fino a nuovo RUN) la loro definizione

230 il computer sceglie la propria strategia di gioco (vedi 790)

240: il mazziere viene variato alternativamente di partita in partita.

250-480: inizializzazione di alcune delle variabili che al termine di ogni partita perdono il valore originario, viene mischiato il mazzo.

440-460: controllo che i valori delle quattro carte da mettere in tavola a inizio partita siano tutti diversi.

490: stampa indicatore carte residue.

500-550: ciclo di chiamata della procedure che coinvolte dalla distribuzione in tavola delle prime quattro carte (vedi 950, 1040, 810), inizializzazione del vettore che raccoglie le carte di primiera (vedi 2170 e 2360).

560-700: ciclo per la gestione dei turni di gioco.

590-660: distribuzione delle carte ai giocatori, aggiornamento vettore carte andate (solo per quelle date al computer); scelta del carattere per il disegno delle carte "in mano".

670: "abbassamento del mazzo".

680-690: scansione dei turni di gioco e invio alle procedure conseguenti.

710-760: raccolta delle carte rimaste in tavola a partita conclusa e loro assegnazione a chi ha preso per ultimo (vedi 1840).

770: vedi 2070.

780: aggiornamento del vettore usato dal computer per scegliere la strategia.

790: se sconfitto il computer risceglie la strategia (vedi 230).

800-940: routine per la stampa delle carte.

800: eseguita quando è necessario evidenziare lo sfondo della carta da stampare.

810: le carte in tavola hanno tutte caratteri di sfondo di codice ASCII diverso per sveltire le operazioni di evidenziazione.

820-830: calcolo coordinate di stampa.

840: scelta del carattere che rappresenta il valore della carta.

850-870: stampa sfondo carte (o loro cancellazione).

890-930: eventuale stampa valore e seme.

950: aggiornamento vettore carte andate (eseguito qui per le carte in tavola all'inizio e per le carte giocate dall'umano).

960-1030: stampa in tavola della carta, quanto più centralmente possibile.

1000: aggiornamento del totale punti in tavola.

1020: se non si tratta di una delle quattro carte messe in tavola all'inizio, ne viene evidenziato lo sfondo (e per ora non si aggiorna F(0..41); vedi 1200 e 1640).

1040-1130: l'ultima carta messa in tavola non prende, il suo sfondo torna normale, compare sotto di essa la lettera di riferimento, viene aggiornato F(0..41).

1140-1310: l'umano sceglie che carta giocare, se c'è ambiguità sceglie cosa prendere, chiamata delle procedure opportune per concludere il turno di gioco (vedi 1650, 1060, 1780).

1320-1640: il computer sceglie che carta giocare, chiamata delle procedure opportune per concludere il turno di gioco (vedi 1650, 1060, 1790).

1370: valutazione preliminare delle possibilità di presa della carta in esame.

1400: calcolo e assegnazione di un opportuno coefficiente di giocabilità (solo per la carta non prendente o che fa scopa).

1420-1530: esame, per la carte considerata delle combinazioni di presa permesse e scelta tra queste della più conveniente.

1520: se la carta considerata può prendere una carta di uguale valore non si considerano (è una regola) altre eventuali combinazioni di presa.

1560: aggiustamento del coefficiente di giocabilità se il computer ha valutato non opportuno, prendendo, risebiare poi di subire scopa.

```
10 DIM F(4) AB(41  2)) 39 B(39) L 5) S 1  P 1  Q0 1    6,1  2(3 1  C0(3 ,0S 21
  DF 1
20 DATA 0 1 2 01 9 02 4 03 12 5,04 012 13 6 05 013 14 23 7 08 014 025 15 24,8 07
  015 024 16 120 25 34
30 DATA 9 08 016 025 034 17 124 26 35-0)20 8 0 16, ← 236T(<=ABCIJKLQ 3844447C44
4444 0 1 1 12  +03472<)BDEJNHR
40 DATA 38440400)0207C FFFFFFFFFFFFFFFF 2 9,19 $-039=TCEFKNOQ 384404100444308 009
67F7F3E)C08 4 5 14    47B=EBLPR
50 DATA 081820407C8820 0000)C3E7F3E(C08 6,8 15, 28<TSQLN TC407804844430-1030107C
FE5410) 9 12 )8 )6)0FKP
60 DATA )8204278444430 0000)C3E7F7F36 13 )T 21 5:0J0 7C04001000000 A5A5A5A5A5A5A
5A5 10 28 10 (AM 04040404044450
70 DATA FF99FF99FF99FF99 24 3) 18 81 3844444544834 FFFF81FFFFFF81FF,32 41 10 H,
444820505048A4 FF9999FFFF9999FF
80 H$= PUNTEGGIO   VITTORIE   SCOPE   CARTE   ORI   SETTEBELLO
  PRIMIERA
90 T$= FFFFFFFFFFFFFFFFF9F2F3F3F3F9F3F3F0F0F0F0F0F0F0F0F0303030303030303030"
100 I$= TTTTTT8811818101 0FFLLRRFLRLFFLLRRFLRLDHLLYIDG0UKKVVW80I#0K0KGOODCVIVIVISS
5
110 CALL CLEAR
120 CALL SCREEN(5
130 FOR A=0 TO 41
140 READ AB(A)
150 NEXT A
160 FOR A=0 TO 9
170 READ /1 A  Z A  PL A) E# A  C# D0
180 CALL COLOR A+) 16 1)
190 CALL CHAR 112+A  00 &C0)
200 CALL CHAR(134+A  00"&C0)
210 CALL CHAR(184+A D0)
220 NEXT A
230 SG=-(RND*DF(1(-RND*DF 0))=0)
240 G=- G=0)
250 FOR A=0 TO 9
260 H(A)=-1
270 QT A+1()=4
280 FOR B=) TO 4
290 K(A*4+B-1)=A+B/10
300 NEXT B
310 NEXT A
320 CALL COLOR(B 16 1)
330 FOR A=0 TO 1
340 R$(A)=""
350 CALL COLOR 11+A 9 16(
360 CALL COLOR(13+A 2,)6)
370 S(A)=0
380 FOR B=0 TO 39
390 RANDOMIZE
400 C=INT(RND*36)
410 E=K 0)
420 K(0)=K(C)
430 K(C)=E
440 FOR D=37 TO 0
450 IF INT(K(D-1))=INT(K(B))THEN 390
460 NEXT D
470 NEXT B
480 NEXT A
490 CALL VCHAR(10,17 122 6)
500 FOR A=0 TO 3
510 Z A 0)=0
520 . A 1 =0
530 R=K(36+A)
540 GOSUB 900
550 NEXT A
560 FOR U=0 TO 35
570 ST=1
580 IF INT(U/6)<>U/6 THEN 620
590 FOR N=G*5 TO 5-G*5 STEP 1-G*2
600 L(N)=K(U+N)
610 R=L(N)
620 QT(R+1)=QT(R+1)+(N>2)
630 X=)24-(N>2)*7
640 GOSUB 820
650 CALL HCHAR(Y+5 T+1 32-(N<2)*(17+N))
660 NEXT N
670 CALL HCHAR 18+U/6 17 32)
680 G=- G=0
690 ON G+1 GOSUB 1320,1140
700 NEXT U
710 E$=""
720 FOR A=0 TO 9
730 IF H(A)=-1 THEN 750
740 E$=E$&STR$ A
750 NEXT A
760 GOSUB 1840
770 GOSUB 2070
780 DF(SG)=DF(SG)+P(0)+P(1)
790 IF C=1 THEN 230 ELSE 240
800 CALL CHAR(33+N,D$(
810 X=33+N
820 Y=ASC SEG$(I$ 57+N 1 )-65
830 T=ASC SEG$(X$ 37+ST+N,1))-65
840 E=112+R- R INT(R)) 2)*22
850 CALL VCHAR(Y T X 5)
860 CALL VCHAR(Y T+1,X 5-(X=32))
870 CALL VCHAR(Y T+2 X 5)
880 IF (X=32 + X=131)THEN 940
890 CALL HCHAR(Y T E)
900 CALL HCHAR Y T+2,E)
910 CALL HCHAR(Y+4 T+2 E)
920 CALL HCHAR(T+4 T E)
930 CALL HCHAR(Y+2 T+1 125+(R-INT(R))*0)
940 RETURN
950 QT(R+)) =QT R+1)-1
960 SI=7
970 FOR Q=0 TO 10
980 IF I Q)THEN 1030
990 I(Q)=R
1000 P'=PV+INT(R)+1
1010 N=Q
1020 ON A(4)+2 GOTO 1040 900
1030 NEXT Q
1040 CALL CHAR(33+N  FFFFFFFFFFFFFFFF")
1050 GOSUB 810
1060 FOR B=1 TO LEN(E$ R))
1070 C=ASC(SEG$(E$(R) B 1))-48
1080 F C =F(C)+1
1090 NEXT B
1100 CALL CHAR(X "FFFFFFFFFFFFFFFF")
1110 H(R)=Q
1120 CALL HCHAR(Y+5,T+1 97+N)
1130 RETURN
1140 CALL SOUND(250 810 1
1150 CALL KEY(0 M LA)
1160 (=M-49
1170 IF (N)-1)- N<3)THEN 1150
1180 IF L(N)=0 THEN 1150
1190 GOSUB 1650
1200 ON CC+1 GOTO 1060 1790,1200
1210 CALL SOUND(390 110,1)
1220 E$=""
1230 T=1
1240 CALL KEY(0 M LA)
1250 CALL SOUND(10 7192 1)
1260 IF LA- M>96)-(M<108)<9 THEN 1240
1270 IF H(I(M-97)=-) THEN 1240
1275 IF POS(E$ STR$(INT(I(M-97  ) 1 THEN 1240
1280 T=T+INT(I(M-97)I+1
1290 E$=E$&STR$(INT(I(M-97)))
1300 IF T=INT(R)+2 THEN 1790
1310 IF T<INT(R)+2 THEN 1240 ELSE 1210
1320 N=G
1330 FOR M=0 TO 2
1340 R=L(M+3)
1350 IF R=0 THEN 1620
1360 Y=INT(R)+1
```

1580 strategia rude. calcolo e assegnazione del coefficiente di giocabilità

1590 il computer giocherà la carta con coefficiente maggiore

1650-1770 la carta giocata sparisce dal lotto delle carte in mano, appare in tavola con sfondo evidenziato, valutazione delle sue possibilità di presa

1780-1820 vengono evidenziate le carte oggetto di presa

1790 l'attributore di presa assume il valore opportuno

1830 la carta giocata prendente viene aggiunta al lotto delle carte oggetto di presa

1840-2060 cancellazione delle carte prese, aggiornamento di F(0 41), assegnazione delle carte stesse a chi ha preso, aggiornamento del totale punti; controllo realizzazione scopa (ed eventualmente memorizzazione della carta che l'ha realizzata, segnale acustico di conferma, aggiornamento del totale punti di scopa)

1860 i numeri rappresentati alfanumericamente in E$ vengono ritrasformati in numeri con l'istruzione ASC più veloce della VAL

2020, 2220, 2380 si tenga presente che i codici del tipo 0 n vengono convertiti alfanumericamente come ".n" (vedi inoltre le istruzioni in 2110, 2240 e 2660)

2070-2450 calcolo e stampa risultato finale, stampa delle carte considerate per l'attribuzione dei punti di scope, ori, settebello, primiera e visualizzazione del totale carte prese dai due giocatori, aggiornamento dei campi del vettore contenente i dati numerici globali

2430 calcolo del vincitore della partita

2440 stampa del risultato finale i due quadratini colorati che vengono stampati ai lati della scritta sono uno bianco e uno rosso in caso di pareggio, altrimenti assumono il colore di riferimento del vincitore (bianco = computer, rosso = umano); si tenga presente che CHR$(x.5) = CHR$(x + 1)

2460-2600 stampa dei dati riassuntivi (voce per voce) per computer e umano valori numerici e linea bicolore di venti caratteri nella quale (lunghezza tratto bianco/lunghezza tratto rosso) = (dato computer/dato umano)

2490 se i dati numerici sono uguali immediata stampa degli stessi e della linea bicolore divisa in due tratti uguali

2520 definizione del pattern del carattere di congiunzione tra il tratto bianco e il tratto rosso della linea in modo che il rapporto (lunghezza singolo tratto/lunghezza totale linea) differisca di meno dello 0 626% (1/160) dal valore del corrispondente rapporto tra i dati numerici

2530 stampa dato numerico computer, linea colorata, dato numerico umano (i dati numerici vengono stampati simmetricamente rispetto agli estremi della linea colorata e possono assumere valori fino a 999 senza provocare irregolarità di stampa).

2560 riga di caratteri bianchi su sfondo nero stampata come separatore dopo i dati relativi alle vittorie, tale riga funge anche da indicatore della strategia adottata dal computer se i caratteri bianchi sono linee, il computer ha giocato in maniera evoluta, se sono quadretti ha giocato in maniera rude

2620-2750 subroutine per la stampa di carte nel dopo partita (vedi 2070 e 820), assegnazione del punto di primiera e stampa del punteggio di primiera

2760-2790 procedura per arrestare l'esecuzione del programma in attesa che venga premuto un tasto con susseguente pulitura dello schermo MC

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 157.

SIEMENS

PT88, si sente vo

. e puoi telefonare in pace, lavorare indisturbato, magari ascoltare musica, leggere una rivista e intanto la tua PT 88 stampa In silenzio
La tecnologia Ink-Jet dà quanto altre stampanti non possono offrire un vero silenzio che si va ad aggiungere alle grandi qualità operative della PT 88
8 set di caratteri nazionali, 150 cps, 80 colonne (132 con la PT 89), 3 densità di scrittura (10, 12, 17 c/"), carta in rotoli, a soffietto e foglio singolo, grafica in bit image e scanner mode compatibile con la grafica

Montecarlo

di Paola Materassi - Firenze

Sulla falsariga dei vari "Pole position" o "Chequered flag", che girano sul Commodore 64, sullo Spectrum, o su altri computer, in questo programma viene simulata una gara di formula 1. Certo il risultato è molto inferiore a quello ottenuto nei succitati programmi, che sfruttano la velocità del linguaggio macchina, in particolare ciò che si vede sullo schermo non rappresenta ciò che un pilota vedrebbe davanti a sé sul cruscotto e attraverso il parabrezza, ma le macchine sono viste dall'alto. Considerando che il programma è scritto in Basic, anche se esteso, il risultato ci sembra più che soddisfacente. L'autore fa ampio uso delle sprite, sfruttando quasi al meglio le possibilità da queste offerte.

Sebbene il titolo del programma si rifaccia al noto circuito cittadino monegasco, notoriamente piuttosto tortuoso, nel nostro caso la pista che compare sullo schermo è sempre dritta, con il prato e due file di alberi ai lati, anche questi visti dall'alto. In pista vi saranno sempre 5 auto di cui una sola pilotata da tastiera. Il colore della macchina del giocatore, manco a dirlo, è rosso, ma dato che non tutti usano un televisore a colori, tale auto viene segnalata prima del via da una freccia lampeggiante. I comandi della vettura sono acceleratore e freno (rispettivamente i tasti S e D) e lo sterzo (tasti E e X), il tasto ALPHA-LOCK deve rimanere premuto. Sotto la pista compaiono una serie di informazioni che, sebbene non siano degne di assurgere al rango di cruscotto, rimangono pur sempre molto utili. Alla sinistra troviamo un tachimetro, seguito da un contachilometri (che funziona per multipli di dieci), un contatempo, un indicatore del numero di urti. Il contatempo viene aggiornato ogni volta

Questo programma è disponibile su cassetta presso la redazione. Vedere l'elenco dei programmi disponibili e le istruzioni per l'acquisto a pag. 157

```
110 !!!!!!!!!!!!!!!!!!!!!!!
120 !  M O N T E C A R L O  !
130 !                       !
140 !!!!!!!!!!!!!!!!!!!!!!!
150
160  gara di formula 1 per
170
180  TI-99/4A extended BAS
190
200  Realizzato da
210
220     Paola Materassi
260
270
280  !!!!!TITOLI!!!!!!
290 CALL CLEAR :: CALL SCREEN(2) :: CALL MAGNIFY(2)
300 CALL COLOR(9,16,16) :: CALL HCHAR(1,1,96,64) :: CALL HCHAR(11,1,96,64)
310 FOR C=1 TO 32 STEP 3 :: CALL COLOR(10,2,2) :: CALL HCHAR(1,C,104) :: CALL HCHA
R(11,C,104) :: NEXT C
320 FOR C=2 TO 32 STEP 3 :: CALL HCHAR(2,C,104) :: CALL HCHAR(12,C,104) :: NEXT C
330 CALL SPRITE(#1,72,5,60,16,#2,79,5,50,40,#3,78,5,40,64,#4,84,5,30,88,#5,69,5,
20,112)
340 CALL SPRITE(#6,67,5,20,136,#7,65,5,30,160,#8,82,5,40,184,#9,76,5,50,208,#10,
79,5,60,232)
350 DISPLAY AT(16,1):"    *   GARA VIDEO PER    *        *  AUTO DI FORMULA 1 *
          PRESS ANY KEY  "
360 CALL KEY(0,K,S) :: IF S=0 THEN 360
370 CALL DELSPRITE(ALL) :: CALL CLEAR :: CALL SCREEN(11) :: CALL COLOR(14,3,3) :: C
ALL COLOR(13,16,16) :: CALL COLOR(12,7,7)
380 FOR R=1 TO 3 :: CALL HCHAR(R,1,112+8*R,32) :: CALL HCHAR(R+21,1,112+8*R,32) ::
NEXT R
390 DISPLAY AT(9,1)BEEP:"NUMERO PARTENTI (1-5)?" :: ACCEPT AT(9,24)SIZE(1)VALIDA
TE(DIGIT):NP
400 DISPLAY AT(12,1)BEEP:"LUNGHEZZA DEL                PERCORSO (KM) ?" :: ACCEPT
AT(14,24)SIZE(3)VALIDATE(DIGIT):LG :: FOR D=5 TO 14
410 DISPLAY AT(D,1) :: NEXT D :: RESTORE 420 :: CALL SCRIVI(12)
420 DATA 5,6,------------------,6,7,COMANDI DI GUIDA,7,6,------------------,10,2
,ACCELERATORE,12,2,FRENO,14,2,STERZO - D)
430 DATA 16,2,STERZO A SX,10,18,TASTO "D",12,18,TASTO "S",14,18,TASTO "X",16,
18,TASTO "E",20,6,< PRESS ANY KEY >
440 DIM NP$(9),P(9),SHP$(9),SP(9) :: IZ :: RI=18 :: CALL KEY(0,K,S) :: IF S=0 THEN
440
450 FOR W=1 TO NP :: DISPLAY AT(RI,2)BEEP:"NOME DEL PILOTA N ";W :: ACCEPT AT(RI
+2,2)VALIDATE(ALPHA):NP$(W) :: CALL CLEAR :: CALL MAGNIFY(3)
460  ! DEFINIZIONE SPRITE !
470 AUTO$="0F0F1OFF467E46FF107C7C000000000000000707F23F0F3FF20707"
480 ROTT$="3E3E10FF627E62FF102E3E000000000000044EE7AA303030AAE74E04"
490 ROTS$="00FBF9000132EE567E5666E0201000F0F000C1C00000050607060706840900001C1C"
500 ANGE$="000C1C3E7EFFE300070F1F3E6EC4004004123070500C0C0C0C0404040404"
510 ALBE$="000F283F5077FFFFFF7C3F7F1E07010000E07EFCECFEDEFBFF6EFEF4DCFBE000"
520 BANS$="3E3E3E0202020203"&RPT$("0",25)&"C0C04FF0F0F0F0F0F0A0A0A0A0A10"
530 BANG$="000000007030".FEFB703"&RPT$("0",12)&"0C041F3F4F0F3F2222620203"
540 CALL CHAR(124,AUTO$,120,ROTT$,132,ALBE$,136,BANS$,140,BANG$,120,ROTS$,116,AN
GE$)
550  ! CARATTERI SPECIALI !
560 RESTORE 570 :: FOR CR=97 TO 97 :: READ CH$ :: CALL CHAR(CR,CH$) :: NEXT CR
570 DATA 000000FFFF1818181818181818,181818,F1F,000001F1F181818,000000F0F81818I
8,1818180F0F,181818FFFF
580 DATA 000000FFFF181818,1818180FFFF181818,181818181F1F181818,181818F8F0181818
590 CALL CHAR(113,"0406FF0604",115,"00E0A0EC08A0") :: CALL COLOR(11,2,15)
600 FOR SET=1 TO 8 :: CALL COLOR(SET,11,5) :: NEXT SET :: CALL COLOR(9,3,3) :: CAL
L COLOR(10,15,15) :: CALL SCREEN(5)
610  !! CAMPO DI GARA !!
620 CALL HCHAR(1,1,104,320) :: CALL HCHAR(4,1,96,96) :: CALL HCHAR(17,1,96,96)
630 DISPLAY AT(1,1)BEEP:"&RPT$("*",26)&"* PILOTA "&"*"&"      "&RPT$("*",
26)&"*"
640 DISPLAY AT(20,1)BEEP:"%%%%%%"&"%%%%%% %%%%%%,%%%%% & KM H &  KM  & TIME & URT
I"
650 DISPLAY AT(22,1):"%%%%%%-%%%%%-%%%%%%-%%%%%& 0  & 0  & 0  & 0 & %
%%,%%+%%%%%%%+%%%%%'+%%%%%%"
660 CALL SPRITE(#2,133,4,33,85,#3,132,8,33,170,#4,132,13,33,255)
670 CALL SPRITE(#5,136,13,130,42,#6,132,13,130,127,#7,132,8,130,212)
680 CALL SPRITE(#9,124,2,56,80,#10,124,11,111,80,#11,124,16,74,80,#12,124,5,93,1
...)
690 CALL SPRITE(#8,124,7,93,80) :: FOR D=1 TO 5 :: DISPLAY AT(13,7)SIZE(1):"*" ::
DISPLAY AT(2,15)SIZE(1) :: CALL SOUND(100,500,0,700,0,-2,0)
700 FOR DD=1 TO 30 :: NEXT DD :: DISPLAY AT(13,7)SIZE(1):" " :: DISPLAY AT(2,11)
SIZE(3) :: DISPLAY AT(3,16)SIZE(1):"*"
710 FOR DD=1 TO 30 :: NEXT DD :: NEXT D
720 RESTORE 730 :: CALL SCRIVI(4)
730 DATA 9,12,PREMERE IL TASTO "0" 12 PER AVERE IL 12,16,SEGNALE DI,14,16,0 PRONT
O !
740 CALL KEY(0,K,S) :: IF S=0 THEN 740
750 CALL HCHAR(8,1,104,224) :: CALL SPRITE(#13,136,2,137,100) :: FOR D=1 TO 700 ::
NEXT D :: CALL SOUND(600,830,0)
760 CALL PATTERN(#13,140) :: FOR D=1 TO 80 :: NEXT D
770  !  !!!  ! A !!!
780 X=-1 :: Y=12 :: 10=-10 :: V(1)=15 :: 12=8 :: URTI,TE=0
790 FOR D=8 TO 12 :: CALL MOTION(#D,0,2) :: NEXT D
800  !     GESTIONE           !! COMANDI DI GUIDA !!
810 CALL KEY(1,K,S) :: CALL COINC(ALL,URTI) :: ... :: IF S=0 THEN 810 ELSE CALL D
ELSPRITE #12
820 CALL KEY(1,K,S) :: CALL COINC(... URTI) :: ... :: IF ... THEN 910 ELSE
```

(continua a pagina 156)

(segue da pagina 155)

```
IF X<=-3 OR X=0 + K=2 + X=3 +!K=5(=0 THEN 830
830 ON K+1 GOTO 840 920 850 860,820 890
840 CALL MOTION(#8 16-X 0): CALL CONTRITE UR GI,X,V( ,K):: CALL MOTION(#8 0 0)::
 GOTO 820
850 X=X 2 :: GOTO 870
860 X=X+1 2 :: IF X<-8 THEN X=-8
870 FOR D=9 TO 12 :: CALL MOTION(#D 0 , D #X+V(D)):: NEXT D :: DISPLAY AT(23,3)S
IZE(4) INT(-X#4)
880 FOR D=2 TO 7 :: CALL MOTION(#D 0 X#15 0) :: NEXT D :: GOTO 820
890 CALL MOTION(#8 16+X 0 :: CALL CONTRITE UR GI,X V(),K):: CALL MOTION(#8 0 0)
:: GOTO 820
900 !*** FINE GARA *****
910 FOR DD=1 TO 6 :: CALL SPRITE(#13,140,2,137 180 :: FOR D=1 TO 30 STEP 5 :: CA
LL SOUND(10,11 5 D :: NEXT D ::
920 CALL PATTERN(#13 136 :: FOR D=1 TO 30 STEP 5 :: CALL SOUND(10,500,D):: NEXT
D :: NEXT DD
930 FOR D=2 TO 12 :: CALL MOTION(#D 0 0):: NEXT D :: FOR D=13 TO 1 STEP -1 :: CA
LL DELSPRITE #D :: CALL SOUND(100 20#D+110,0)
940 FOR DD=1 TO 40 :: NEXT DD :: NEXT D
950 ! CALCOLO PUNTEGGIO !
960 IF GI>1E9 THEN P(V)=0 ELSE P(V)=1000-(20#UR)-TE+LG :: IF P(V)<0 THEN P(V)=1
970 DISPLAY AT(10,5)SIZE(20):"PUNTEGGIO REALIZZATO" :: DISPLAY AT(12 13)BEEP SIZ
E(4):P(V) :: DISPLAY AT(15,7)SIZE(17):"< PRESS ANY KEY >"
980 CALL KEY(0,K,S):: IF S=0 THEN 980
990 FOR D=1 TO 3 :: DISPLAY AT(D,1) :: NEXT D :: FOR D=20 TO 24 :: DISPLAY AT(D,1
) :: NEXT D :: CALL HCHAR(10,1 104,192):: RI=21 :: NEXT V
1000 !***** FINALE *****
1010 DISPLAY AT(8 1):"    &&  &&&&&  &&  &&&"("  &  &    &    &  &  &  &":"
  &     &  &  &  &"    &     &    &  &  &  &"
1020 DISPLAY AT(12 1):"      &    &    &  &  &&&"("    &    &    &  &  &":"
&  &    &    &  &  &"    &&    &    &&  &"
1030 FOR D=1 TO 400 :: NEXT D
1040 !*  ORDINAMENTO  *      *  CLASSIFICA  *
1050 IF NP=1 THEN SP(1)=P(1) :: SNP$(1)=NP$(1):: GOTO 1110
1060 FOR V=1 TO NP :: IND=1 :: M=P(1)
1070 FOR D=1 TO NP-1 :: MAS=MAX(M,P(D+1)) :: IF M<>MAS THEN IND=D+1 :: M=MAS
1080 NEXT D
1090 SP(V)=P(IND):: SNP$(V)=NP$(IND):: P(IND)=-1E8
1100 NEXT V
1110 CALL DELSPRITE(ALL):: CALL CLEAR :: DISPLAY AT(1,6)BEEP:"CLASSIFICA  FINALE
"
1120 DISPLAY AT(3,1):"!%%,%%%%%%%%%%%%%,%%%%%%%%%)"
1130 FOR D=1 TO NP :: DISPLAY AT(4+2#(D-1) 1):"&  &          &         & %%-%
%%%%%%%%%%%-%%%%%%%%/" :: NEXT D
1140 DISPLAY AT(5+2#(NP-1),1):" %%+%%%%%%%%%%%%%+%%%%%%%%%0"
1150 FOR D=1 TO NP :: DISPLAY AT(4+2#(D-1),2)SIZE(2):D :: NEXT D
1160 FOR D=1 TO NP :: DISPLAY AT(4+2#(D-1),6)SIZE(11):SNP$(D):: NEXT D
1170 FOR D=1 TO NP :: DISPLAY AT(4+2#(D-1),19)SIZE(9):"PUNTI";SP(D):: NEXT D
1180 FOR D=1 TO 400 :: NEXT D
1190 DISPLAY AT(7+2#(NP-1),2):"COMPUTER TEXAS INSTRUMENTS"
1200 DISPLAY AT(1,28):"q" :: ACCEPT AT(1,28)SIZE(-1):B$ :: DISPLAY AT(7+2#(NP-1)
-1)BEEP:"   < VUOI RIPARTIRE ? >"
1210 CALL KEY(0,K S):: IF S=0 THEN 1210
1220 IF K=83 THEN CALL CLEAR :: CALL CHARSET :: GOTO 370
1230 IF K<>78 THEN 1210
1240 CALL CLEAR :: STOP
1250 !
1260 !    **************
1270 !
1280 !*  SOTTOPROGRAMMI  *
1290 !
1300 !    **************
1310 !
1320 SUB SCRIV1(N)
1330 FOR D=1 TO N :: READ R,C,OP$ :: L=LEN(OP$) :: DISPLAY AT(R,C)SIZE(L):OP$ ::
NEXT D
1340 SUBEND
1350 !
1360 !
1370 SUB CONTRITE UR GI X V(),K):: TE=TE+1 :: CALL COINC(#2,33,102,2-X,CG):: GI=
GI-CG :: DISPLAY AT(23 10)SIZE(4):GI#10
1380 CALL COINC(ALL,C):: IF C<>0 THEN CALL CRAS(TE,UR,GI X,V(),K)
1390 SUBEND
1400 !
1410 !
1420 SUB CRAS(TE,UR,GI,X,V(),K)
1430 IF X<-3 THEN CALL PATTERN(#8,120)ELSE CALL PATTERN(#8,128)
1440 CALL SOUND(1000,-7,0):: FOR D=2 TO 8 :: CALL MOTION(#D,0 0):: NEXT D
1450 FOR D=9 TO 12 :: CALL MOTION(#D,0,V(D)#-X) :: NEXT D
1460 FOR D=1 TO 6 :: CALL SPRITE(#13 140,2,137 180) :: FOR DD=1 TO 40 :: NEXT DD
:: CALL PATTERN(#13,136):: FOR DD=1 TO 40 :: NEXT DD :: NEXT D
1470 FOR D=9 TO 12 :: CALL MOTION(#D,0,0) :: NEXT D :: CALL COINC(ALL,C)
1480 IF C<>0 THEN CALL MOTION(#8,4 0,#13,0 4):: CALL MOTION(#8 0,0 #13 0 0 :: GO
TO 1470
1490 UR=UR+1 :: TE=TE+10 :: DISPLAY AT(23 4)SIZE(4):"0" :: DISPLAY AT(23,17)SIZE
(4):TE :: DISPLAY AT(23,24)SIZE(5):UR
1500 IF X<-3 THEN CALL DEAT :: GI=1E9 :: GOTO 1520
1510 X=-1 :: CALL PATTERN(#8 124):: CALL SOUND(500,110,5):: CALL PATTERN(#13,140
):: FOR D=1 TO 500 :: NEXT D :: CALL DELSPRITE(#13)
1520 K=-1 :: SUBEND
1530 !
1540 !
1550 SUB DEAT
1560 FOR D=1 TO 100 :: NEXT D :: CALL POSITION(#8 YP,XP) :: CALL SPRITE(#1 116,16
,YP,XP,-5,0)
1570 RESTORE 1580 :: FOR D=1 TO 33 :: READ FR :: CALL SOUND(140 FR 0) :: NEXT D
1580 DATA 165,165,165,44000,165 165,165 44000 165 44000,165 165,44000 44000 196,
196 44000,185
1590 DATA 44000,185 44000 44000 165,44000 165 44000 44000,147,44000,165,185 165
165
1600 FOR D=1 TO 350 :: NEXT D :: CALL DELSPRITE(#1):: CALL HCHAR(20,1,30-160):: 
CALL SCRIV1(3)
1610 DATA 21,2,LA TUA ANDATURA ERA TROPPO,22 2 VELOCE : PURTROPPO NELLO 23,2,5
CONTRO SEI DECEDUTO !!
1620 FOR D=1 TO 500 :: NEXT D :: SUBEND
1630 !
1640 !
1650 END
```

che vi è un urto e non viene aggiornato alla fine della gara. Il valore finale del tempo impiegato sul percorso va a influenzare direttamente il punteggio del giocatore, che è funzione anche del numero di incidenti subiti. Un incidente particolarmente grave,come un tamponamento a velocità troppo elevata, provoca la messa fuori combattimento del giocatore e il corrispondente punteggio viene azzerato.

Possono gareggiare da uno a nove piloti, uno per volta. Per prima cosa il programma chiede la lunghezza in chilometri della corsa, a questa richiesta conviene rispondere con un valore multiplo di 10. Prima di ogni gara il programma chiede il nome del concorrente, al termine dell'ultima gara viene stampata la classifica con i migliori punteggi conseguiti.

Principali variabili utilizzate dal programma

NP: numero piloti (da 0 a 9)
LG: lunghezza gara (da 0 a 999 km)
NP$(9): vettore con i nomi dei piloti
SNP$(9): vettore con i nomi dei piloti ma in ordine di punteggio
PS(9): punteggi realizzati
SP$(9): punteggi in ordine decrescente
V(12): vettore delle velocità iniziali delle auto
X: parametro di velocità
GI: giri fatti (1 giro = 10 km)
UR: urti
te: tempo
W: contatore partenze

MC

Inviate i vostri programmi

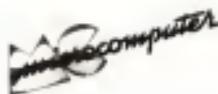
Alcuni lettori ci chiedono, nelle loro lettere, come sottoporre i loro programmi a MC.

È semplicissimo: registrate i vostri lavori su cassetta o disco (se il programma è proprio molto corto può bastare il semplice listato: certo, la cassetta non guasta mai...), corredateli dell'opportuna documentazione e spedite il tutto alla redazione, indicando magari sulla busta la rubrica interessata.

Tutti i programmi che arrivano sono esaminati ed i migliori pubblicati.

Purtroppo non possiamo restituire, per ragioni organizzative, il materiale che ci viene inviato, anche in caso di mancata pubblicazione.

Ricordatevi che migliore è la documentazione, maggiore è la possibilità che il vostro lavoro venga pubblicato: spiegate quindi chiaramente il funzionamento del programma ed accludete tutto quello che pensate possa essere utile (elenco variabili e via dicendo). Soprattutto non dimenticate di indicare il vostro nome ed indirizzo (qualche volta succede!) e, se possibile, il numero telefonico.

Ah, quasi dimenticavamo: naturalmente è previsto un compenso, che varia normalmente tra le 30 e le 100 000 lire, a seconda della qualità del lavoro inviato.

Elenco del software disponibile su cassetta o minifloppy

Per ovviare alle difficoltà incontrate da molti lettori nella digitazione dei listati pubblicati nelle varie rubriche di software sulla rivista, MCmicrocomputer mette a disposizione i programmi più significativi direttamente su supporto magnetico. Riepiloghiamo qui a fianco i programmi disponibili per le varie macchine, ricordando che i titoli non sono previsti per computer diversi da quelli indicati.
Il numero della rivista su cui viene descritto ciascun programma è riportato nell'apposita colonna, consigliamo gli interessati di procurarsi i relativi numeri arretrati, eventualmente rivolgendosi al nostro Servizio Arretrati utilizzando il tagliando pubblicato in fondo alla rivista.

Per l'ordinazione inviare l'importo (a mezzo assegno, c/c o vaglia postale) alla Technimedia srl, Via Valsolda 135, 00141 Roma

Le cassette utilizzate sono Basf C-60 Compusette II, i minifloppy sono Basf singola faccia singola densità

Codice	Titolo programma	MC n	Prezzo	Note
	APPLE II			
DA2/00	Shape Tablet	22	15000	
DA2/01	Motomuro	26	15000	
DA2/02	&DEBUG	28	15000	
DA2/03	EDIT + INPUT	29	15000	
DA2/04	Basic modulare	34	15000	
DA2/05	ANNA Animation Lang	35/37	15000	
DA2/06	Miniset + Leva-DOS	37	15000	
DA2/07	27 programmi grafici	38	30000	
DA2/08	Adventure Editor	38	15000	
	COMMODORE 64			
C64/01	Briscola	25	17000	
C64/02	Serpentone	29	17000	
C64/03	Othello	29	17000	
C64/04	Chase	33	17000	
C64/05	Spreadsheet	34	30000	
C64/06	Bilancio familiare	35	17000	
C64/07	The dark wood	36	17000	
C64/08	Totocalcio sis rid	37	17000	
C64/09	Orchetea	37	17000	
C64/10	Wordprocessor	38	17000	
C64/11	Helicopt	38	17000	
C64/12	Finestra grafica	39	17000	
C64/13	Paroliamo	39	17000	
D64/01	Spreadsheet	34	15000	
D64/02	ADP Basic	da 35 a 39	15000	
D64/03	Wordprocessor	38	15000	
D64/04	Paroliamo	39	15000	
	COMMODORE VIC-20			
CVC/01	VIC-Maze	19	17000	Config base
CVC/02	Pac-Man	23	17000	Config base
CVC/03	Briscola	25	17000	Config base
CVC/04	Grand Prix	28	17000	Config base
CVC/05	Frogger	26	17000	RAM almeno + 3 K
CVC/06	Invaders	29	23000	RAM + 16 K
CVC/07	Othello	29	17000	RAM + 16 K
CVC/08	SKI	31	17000	Config base
CVC/09	VIC-quiz	32	17000	RAM almeno + 8 K
CVC/10	Zigurat	33	17000	Config base
CVC/11	Extended Basic	36	17000	RAM + 16 K
CVC/12	Fireman	36	17000	Config base
CVC/13	Accordi per chitarra	39	17000	RAM almeno + K
CVC/14	Piramide di Iunnuh	39	17000	RAM almeno + K
DVC/01	EXMA	27/28	15000	RAM + 16 K
	SINCLAIR SPECTRUM			
CSS/01	TRILAB	28	17000	
CSS/02	SET di caratteri	27/29	17000	
CSS/03	Grafica TREDIM	29	17000	
CSS/04	Ippica	30	17000	
CSS/05	Graphic-Comp	32	17000	
CSS/06	Macchina del tempo	34	17000	
CSS/07	Piramide di Iunnuh	35	17000	
CSS/08	Over Basic	37	17000	
CSS/09	Prospettiva	38	17000	
CSS/10	Motomuro	39	17000	48 K RAM
	TEXAS TI-99/4A			
CT9/01	Macchina del tempo	27	17000	
CT9/02	Simon	29	17000	
CT9/03	Babilonia	30	17000	
CT9/04	Labirinto 3D	31	17000	
CT9/05	Piramide di Imnnuh	33	17000	Extended Basic
CT9/06	Scrabble	34	17000	
CT9/07	Morphy	35	17000	
CT9/08	Equo canone	37	17000	
CT9/09	Scopa	39	17000	
CT9/10	Montecarlo	39	17000	Extended Basic

Nota
l'iniziale del codice e' C per le cassette, D per i minifloppy

PC TEXT

di Fabrizio Canonica
Arbedo (CH-TI)

Questo programma, scritto interamente in linguaggio macchina, permette di trattare i testi con il PC 1500.

PC TEXT gestisce 36 caratteri per riga, permette inserimenti, cancellazioni, salti di parola in parola, di frase in frase, di riga in riga e naturalmente di lettera in lettera. Tutto questo è possibile sia in avanti che indietro. Il programma consente inoltre cambiamenti di colore in qualsiasi istante, di andare a capo a seconda delle proprie necessità e, durante la fase di stampa, provvede a non tagliare a metà una parola (wrap-around).

Lanciato il programma, con CALL &78C5 o &40C5, il display si cancellerà e il computer si metterà in attesa.

A questo punto ci sono due possibilità.

— Premere SHIFT+CL. Questa operazione cancella un eventuale testo già presente nella memoria disponibile all'utente.

— Cominciare a scrivere il testo. Tutto ciò che si scrive appare sul display che scorrerà a sinistra man mano che si continua con le inserzioni.

Il testo si snoderà quindi su un'unica lunghissima riga. Il programma è comunque dotato di funzioni speciali che permettono il controllo e l'editing del testo scritto. Ecco l'elenco di queste funzioni.

— pressione sul tasto F1 (!) si salta di una parola in avanti,

— pressione sul tasto F2 (") si salta di una parola indietro,

— pressione sul tasto F3 (#) si salta di una frase in avanti,

— pressione sul tasto F4 ($) si salta di una frase indietro.

Il programma considera come frasi tutte quelle che terminano con . (punto), ! (punto esclamativo), ? (punto interrogativo)

— Pressione sul tasto ▼ si salta di una riga in avanti,

— pressione sul tasto ▲ si salta di una riga indietro.

Il programma considera come riga ogni nostra pressione sul tasto ENTER. Sul display è raffigurato col carattere CHR$ 255.

— Pressione sul tasto ◀ ci si sposta di una lettera verso sinistra,

— pressione sul tasto ▶ ci si sposta di una lettera verso destra,

— pressione sul tasto SHIFT+ ◀ equivale al normale DELETE,

— pressione sul tasto SHIFT+ ▶ equivale al normale INSERT, ma attenzione gli spazi creati devono essere riempiti, o eliminati con DELETE, altrimenti, durante la stampa, creeranno altrettanti spazi.

Queste sono le funzioni riguardanti l'editing, ce ne sono però delle altre:

— pressione sul tasto F5 (%) serve per cambiare colore durante la stampa. Sul display apparirà il carattere CHR$ &7E che dovrà essere seguito dal numero del colore desiderato (da 0 a 3),

— pressione sul tasto ENTER, provoca un ritorno a capo forzato durante la stampa. Come già detto, sul display apparirà il carattere CHR$ 255.

Ovviamente durante la stampa questi caratteri non figureranno. Attenzione però: la pressione di questi due tasti dovrà essere OBBLIGATORIAMENTE preceduta da uno spazio (pressione sul tasto SPACE).

— Premendo SHIFT+CL si ha un ritorno al BASIC,

— premendo il tasto F6 (&) si ritorna alla situazione di inizio programma.

Una volta finito di scrivere il testo, per passare alla stampa bisognerà premere il tasto ⬍.

Per l'introduzione del programma consiglio l'uso del PC MON apparso sul numero 23 di MC, che permette l'introduzione dei numeri direttamente in esadecimale, con un notevole risparmio di tempo e fatica.

Attenzione: il programma pubblicato è per la versione 8K.

Per la versione 4K e standard bisognerà cambiare i numeri cerchiati con i seguenti valori:

&3A &42
&3B &43
&5F &57 (&47 per la versione standard).

Il programma è lungo 761 passi. I checksum valgono:

94273 per la versione standard
94305 per la versione 4K
94157 per la versione 8K

Con la versione standard si possono scrivere fino a 1086 caratteri, con la versione 4K 5184 caratteri, con la versione 8K 9280. Eventuali programmi in Basic residenti in memoria verranno inesorabilmente cancellati.

Descrizione programma

&38C5-&3949 accetta, controlla e visualizza il carattere premuto
&3959-&3982 routine di spostamento di una lettera in avanti
&3983-&39AC routine di spostamento di una lettera indietro
&39AD-&39C0 routine cancellamento testo
&39C1-&39ED INSERT
&340A-&3A2F DELETE
&3A3A-&3A5C routine di spostamento di una parola in avanti
&3A5D-&3A79 routine di spostamento di una parola indietro
&3A7A-&3AA1 routine di spostamento di una frase in avanti
&3AA2-&3AC6 routine di spostamento di una frase indietro
&3AC7-&3AE6 routine di spostamento di una riga in avanti
&3AE7-&3B03 routine di spostamento di una riga indietro
&3B04-&3BB0 routine di stampa, con controllo colore, ritorno a capo e wrap-around
&3BB1-&3BBD ricerca dell'indirizzo più alto dove è stato scritto un carattere

Routine impiegate

EE71 cancella il display
E243 accetta un carattere da tastiera, e pone il valore ASCII nell'accumulatore
ED00 visualizza un testo sul display con lunghezza nell'accumulatore e indirizzo di partenza nel registro HL
E669 equivale a BEEP 1
A519 viene scelto il colore posto in E
B4F4 stampa n caratteri. Il numero dei caratteri deve venir posto in E, e l'indirizzo di partenza in HL
A9F1 genera un ritorno a capo. Prima della chiamata HL deve contenere &7B7F. MC

PC TEXT

```
38C5: 8E EE 71 FD
38C9: 48 84 AE 78
38CD: C2 04 AE 78
38D1: C3 68 3B 6A
38D5: C0 FD 28 FD
38D9: A8 FD 88 FD
38DD: 5A 8E E2 43
38E1: B7 1A 8B C8
38E5: BE 08 B5 00
38E9: AE 78 75 8E
38ED: E2 43 B7 1A
38F1: BB 5B B7 08
38F5: BB 8C B7 0C
38F9: 8B 5E B7 1C
38FD: 8B C2 B7 1D
3901: BB EB B7 11
3905: 8B EB B7 12
3909: BB EB B7 13
390D: 8B EB B7 14
3911: 8B EB B7 0D
3915: 89 02 B5 FF
3919: B7 0A 8B E5
391D: 87 0B 8B E5
3921: B7 15 89 02
3925: B5 7C 87 16
3929: 98 66 B7 09
392D: 89 05 85 7E
3931: BA 3A 59 FD
3935: 0A FD 2A 0E
3939: 4C 3B 89 02
393D: 4E DA B1 01
3941: 64 44 FD A8
3945: FD 88 B5 1A
3949: BE ED 00 9E
394D: 67 A5 78 C2
3951: 08 A5 78 C3
3955: 0A FD 4E 9A
3959: FD 0A FD 2A
395D: 4C 5F 89 02
3961: 4E FF 81 09
3965: FD AB FD 88
3969: BE E6 69 9E
396D: 87 44 4C 3B
3971: 89 02 4E DA
3975: 81 01 64 FD
3979: A8 FD 88 B5
397D: 1A 8E ED 00
3981: 9E 9C FD 0A
3985: FD 2A 4C 3B
3989: 89 02 4E C1
398D: 83 09 FD A8
3991: FD 88 8E E6
3995: 69 9E 81 46
3999: 6C 3B 89 02
399D: 6E C0 8B 01
39A1: 66 FD A8 FD
39A5: 88 B5 1A BE
39A9: ED 00 9E C6
39AD: 85 00 68 3B
39B1: 6A C0 61 6C
39B5: 5F 89 02 6E
39B9: FF 99 09 BE
39BD: E6 69 9E DA
39C1: BE 3B B1 FD
39C5: 1B FD 6A 54
39C9: FD 0A 46 67
39CD: 53 A4 86 89
39D1: 02 24 06 99
39D5: 0A 44 FD 88
39D9: A5 78 C0 18
39DD: A5 78 C1 1A
39E1: B5 27 0E FD
39E5: 0A FD 2A FD
39E9: A8 FD 88 9E
39ED: A7 B7 1D 8B
39F1: 18 B7 11 BB
39F5: 3E B7 12 8B
39F9: 63 B7 13 8B
39FD: 7C 87 14 8B
3A01: A0 B7 0A 8B
3A05: C1 B7 0B 88
3A09: DD BE 3B B1
3A0D: 54 FD 18 FD
3A11: 6A FD 0A FD
3A15: B8 FD 5A 44
3A19: F5 A4 86 89
3A1D: 02 24 06 99
3A21: 09 85 00 2E
3A25: FD 0A FD 2A
3A29: FD A8 FD 88
3A2D: BA 39 47 87
3A31: 7E 9B E6 8E
3A35: 3B B1 FD 0A
3A39: FD 2A 84 96
3A3D: 89 02 04 16
3A41: B3 0F 4C 3B
3A45: 89 02 4E DA
3A49: B1 01 64 44
3A4D: 05 87 20 99
3A51: 17 FD A8 FD
3A55: B8 8A 39 47
3A59: B7 7E 8B A7
3A5D: FD 0A FD 2A
3A61: 4C 3B 89 02
3A65: 4E C1 91 17
3A69: 6C 3B B9 02
3A6D: 6E C1 81 01
3A71: 66 46 05 87
3A75: 70 99 17 9E
3A79: 28 BE 3B 81
3A7D: FD 0A FD 2A
3A81: 84 96 89 02
3A85: 04 16 93 37
3A89: 4C 3B 89 02
3A8D: 4E DA 81 01
3A91: 64 44 05 87
3A95: 2E 9B 46 B7
3A99: 3F 9B 4A 87
3A9D: 21 9B 4E 9E
3AA1: 21 FD 0A FD
3AA5: 2A 4C 3B 89
3AA9: 02 4E C1 91
3AAD: 5C 6C 3B B9
3AB1: 02 6E C1 81
3AB5: 01 66 46 05
3AB9: 87 2E 9B 6B
3ABD: B7 3F 98 6F
3AC1: B7 21 98 73
3AC5: 9E 21 BE 3B
3AC9: 81 FD 0A FD
3ACD: 2A 84 96 89
3AD1: 02 04 16 93
3AD5: 84 4C 3B 89
3AD9: 02 4E DA 81
3ADD: 01 64 44 05
3AE1: B7 FF 99 17
3AE5: 9E 95 FD 0A
3AE9: FD 2A 4C 3B
3AED: 89 02 4E C1
3AF1: 91 A1 6C 3B
3AF5: 89 02 6E C1
3AF9: 81 01 66 46
3AFD: 05 B7 FF 99
3B01: 17 9E B2 BE
3B05: 3B 81 B5 20
3B09: 51 85 7E 1E
3B0D: B5 01 AE 79
3B11: F4 B5 24 AF
3B15: 78 C4 48 3B
3B19: 4A C0 FD 6A
3B1D: 58 00 5A 00
3B21: 25 B7 7C 8B
3B25: 18 87 FF 8B
3B29: 57 87 7E 9B
3B2D: FE B7 20 88
3B31: 04 64 54 9E
3B35: 15 14 A7 78
3B39: C4 81 20 8E
3B3D: 5D 64 44 FD
3B41: AB FD 8B 25
3B45: FB B1 30 B7
3B49: 04 93 2B 2A
3B4D: BE A5 19 FD
3B51: 0A FD 2A 44
3B55: 64 58 00 5A
3B59: 00 9E 38 FD
3B5D: AB FD 88 50
3B61: 14 AE 78 C5
3B65: A5 78 C4 F8
3B69: A1 78 C5 AE
3B6D: 78 C4 14 2A
3B71: 8E B4 F4 FD
3B75: 0A FD 2A 58
3B79: 00 5A 00 64
3B7D: FD 28 9E 60
3B81: FD 6A B5 24
3B85: AE 78 C4 FD
3B89: A8 FD 88 48
3B8D: 7B 4A 7F 8E
3B91: A9 F1 FD 0A
3B95: FD 2A 44 64
3B99: 9E 7A B5 24
3B9D: AE 78 C4 FD
3BA1: A8 FD BB 48
3BA5: 7B 4A 7F BE
3BA9: A9 F1 FD 0A
3BAD: FD 2A 9E 55
3BB1: 48 3B 4A C0
3BB5: 45 B7 00 99
3BB9: 05 FD 5A 56
3BBD: 9A 8A FD 00
```

Hex dump del programma PC TEXT (per i byte cerchiati vedi testo).

Tutti i numeri
che occorrono
per essere il n. 1
delle stampanti in Italia
1 Operare solo nel mercato delle periferiche
2 Produrre più di 350.000 macchine all'anno
3 Offrire una vasta gamma di modelli
4 Essere presente nelle varie fasce di mercato
5 Avere oltre 10 anni di attività in Italia
6 Investire in ricerche tecnologiche per anticipare le esigenze del mercato
7 Garantire una efficace assistenza tecnica su tutto il territorio nazionale
Guarda caso! La
MANNESMANN
TALLY
ha tutti i numeri... dal 1 al 7
adverteam

L'istruzione PRINT USING

In questo numero ci occuperemo dell'istruzione "PRINT USING" andandone ad analizzare le caratteristiche, che forse non tutti conoscono, magari perché non espressamente citate nel manuale allegato al personal computer.

Innanzitutto vediamo la sintassi dell'istruzione in oggetto:

PRINT USING <espr. di tipo stringa>, <lista di espress.>

dove

"<espr. di tipo stringa>"

è una stringa opportuna (calcolata oppure esplicita), che consente la formattazione di ciò che è indicato con "<lista di espress.>".

Generalmente si può dire che, rispetto ad una semplice PRINT, la PRINT USING introduce come detto una certa formattazione dell'output, intendendo con questo termine l'output di valori numerici con un numero prefissato di cifre prima e dopo la virgola e preceduti o seguiti da appositi simboli oppure l'output di parti di stringhe di caratteri.

Il tutto a seconda del contenuto della "<espr. di tipo stringa>" definibile a piacere.

Analizziamo dapprima la formattazione delle stringhe.

La formattazione dei campi alfanumerici

Quando si devono stampare delle stringhe alfanumeriche, le si possono formattare con tre caratteri speciali: vediamo ora i singoli caratteri, un esempio di linea di programma e l'output corrispondente.

— **carattere "!"**: indica che della stringa data deve essere stampato solo il primo carattere. Con

PRINT USING "!";"MCmicrocomputer","Audio Review"

si ottiene in uscita

MA

e cioè solo le iniziali delle stringhe successive.

— **stringa di "/n spazi/"**: indica che della stringa data devono essere stampati 2+n caratteri. Se tra le due "backslash" non si mettono spazi (n=0) allora verranno stampati i primi due caratteri. Se la stringa è più corta del campo risultante, allora i caratteri stampati vengono giustificati a sinistra, cioè con l'inserimento di spazi bianchi a destra. Ad esempio,

PRINT USING "//";"MCmicrocomputer"

dà in uscita

MC

mentre l'istruzione (con 5 spazi tra le "backslash")

PRINT USING "/   /";"MC microcomputer"

fornisce in uscita

MC micr

e cioè 7 caratteri della stringa data.

— **carattere "&"**: indica che deve essere stampata correttamente la stringa data e cioè con

PRINT USING "&","Audio Review" si ottiene in uscita

Audio Review

e cioè la stringa completa.

Mentre i primi due tipi di formattazione non danno problemi, l'ultimo carattere ("&") può destare qualche perplessità, in quanto apparentemente sembrerebbe inutile.

Invece può essere usato, anzi deve, laddove vogliamo eliminare l'effetto USING dovendo usare però una subroutine di stampa contenente tale istruzione.

Supponiamo ad esempio di avere in un programma una subroutine di stampa tra le cui istruzioni compare una

PRINT USING F$,A$,B$,C$

dove F$ è la stringa di formattazione contenuta per l'appunto in una variabile e perciò preimpostabile prima della chiamata della subroutine.

Supponendo perciò che ad un certo punto vogliamo stampare le stringhe A$, B$ e C$ così come sono, ecco che ci torna utilissimo porre

F$ = "&"

prima della chiamata della subroutine stessa.

Un esempio di applicazione potrebbe essere la gestione di tabulati o prospetti dove la lunghezza di un campo alfanumerico potrebbe variare da una colonna all'altra del tabulato.

Stringha di formattaziona dell'istruziona PRINT USING

Stringa	Effetto nell'output
!	Primo carattere delle stringhe
\n spazi\	Primi n+2 caratteri delle stringhe
&	Stringhe complete
#	Posto per una cifra
.	Posto per il punto decimale
+	Segno del numero (a destra o a sinistra)
—	Segno "—" a sinistra se necessario
**	Asterischi prima del numero
$$	Segno del dollaro prima del numero
**$	Asterischi e dollaro prima del numero
,	Virgole delle migliaia, milioni ecc
^ ^ ^ ^	Notazione esponenziale
—	Stampa del carattere successivo

Tabella 1 - Effetto della stringa di formattazione sull'output di stringhe (primi tre casi) e di numeri (tutti i successivi). Per i dettagli rimandiamo senz'altro all'articolo dove sono riportati vari esempi.

La formattazione di campi numerici

Quando, analogamente al caso delle stringhe, bisogna stampare dei valori numerici, con un ben preciso numero di cifre per problemi di incolonnamento o con inserzione automatica di speciali simboli, ecco che si possono usare fino a dieci tipi di stringhe di formattazione, in genere formate da uno, due o più caratteri.

Anche in questo caso indicheremo la stringa di formattazione ed un esempio di istruzione con il relativo output.

— **carattere "#"** - Il simbolo "#" (chiamato a seconda dei gusti "diesis", "cancelletto", "canale", "number", ecc.) è usato per indicare la singola posizione di stampa di una cifra. Nel caso in cui il numero da stampare possiede meno cifre di quelle specificate dai "diesis", allora tali cifre vengono (correttamente!) giustificate a destra e cioè precedute da un numero opportuno di spazi bianchi.

È questa senz'altro l'opzione della PRINT USING più conosciuta dai programmatori e utilizzata appunto per incolonnare numeri ottenendo il naturale incolonnamento a destra in contrapposizione all'innaturale incolonnamento a sinistra proprio dei computer.

— **carattere "."** - Il "punto" può essere inserito in una qualsiasi posizione di un campo numerico, seguito e preceduto da un numero desiderato di "diesis". In questo caso l'incolonnamento avverrà tenendo conto delle cifre decimali. Una cosa che forse non tutti sanno è che, se necessario, i valori decimali vengono arrotondati, inoltre, se il valore non ha la parte intera, allora viene stampato comunque uno "0" seguito dal punto decimale e dalle cifre prestabilite.

Ad esempio il programmino

```
10 A$ = "##.##"
20 PRINT USING A$ 31.41
30 PRINT USING A$, 1.3
40 PRINT USING A$, 5.119
50 PRINT USING A$, .55674
```

darà in uscita la seguente tabella di valori

```
31.41
 1.30
 5.12
 0.56
```

dove, negli ultimi due casi si è avuto un arrotondamento del valore numerico.

— **carattere "+"** - Il segno di addizione, posto all'inizio o alla fine della stringa

di formattazione (costituita ad esempio da "diesis" e " ") fa si che corrispondentemente venga stampato, prima o dopo il numero, il suo segno positivo o negativo che sia (attenzione!)

Ad esempio

```
PRINT USING "+## #", —11.33
```

fornisce in uscita

```
—11 +0 3
```

— **carattere "—"** - Il segno "meno", a differenza del precedente, impone la stampa di un segno "meno" (solo se il numero è negativo) subito dopo il numero stesso Strano, no?! Ad esempio l' istruzione (il "—" va posto in fondo)

```
PRINT USING "## ##—", —11 33
```

da un output del tipo

```
1 10— 0 33
```

— **stringa "**"** — Un doppio asterisco all'inizio di un campo di formattazione forza la stampa di asterischi al posto di eventuali spazi, nel caso che il numero sia formato da meno cifre che non il campo prefissato I due stessi asterischi indicano, ove necessario, la posizione per altre due cifre Ma vediamo degli esempi

Con l'istruzione

```
PRINT USING "**## #" 3 14,—77 855
```

otterremo

```
**3 1 *—0 8 855 0
```

— **stringa "$$"** - Il doppio dollaro consente di stampare "prima" del numero formattato il carattere "dollaro", considerando che anche in questo caso i "due dollari" indicano due ulteriori cifre del campo di formattazione

L'istruzione

```
PRINT USING "$$### #",123 44
```

fa stampare in uscita

```
$123 4
```

— **stringa "**$"** L'insieme di due asterischi e del dollaro all'inizio della stringa di formattazione fa si che si ottenga l'effetto combinato delle due precedenti stringhe e cioè il riempimento con asterischi e la stampa del singolo "dollaro" subito a sinistra del numero formattato Per non complicarci ulteriormente le idee lasciamo agli abili lettori la sperimentazione di questa stringa

— **carattere ","** — Siamo arrivati alla "fatidica" virgola, che gli anglosassoni si ostinano ad usare, insieme al punto, in maniera esattamente opposta al nostro modo In pratica ponendo una virgola subito a sinistra del "punto decimale" nella stringa di formattazione si ottiene la stampa automatica di una virgola separatrice delle migliaia dei milioni, ecc Se invece la virgola viene posta alla fine della stringa di formattazione, allora verrà stampata alla destra del numero, come parte integrante del numero stesso

Ad esempio

```
PRINT USING "####, ## ,5555 5
```

fornisce in uscita il valore

```
5,555 50
```

— **stringa "^^^^"** - La stringa formata da quattro caratteri "accento circonflesso" o "freccia in su" specificano, se posti alla destra della stringa di formattazione, che il numero verrà stampato in formato esponenziale In particolare i quattro simboli "^" (in inglese "carat") specificano che verrà stampata la stringa "E+xx" dove "xx" è l'esponente calcolato correttamente in base alle posizioni delle cifre intere e decimale stabilite dalla solita stringa di formattazione Ancora una volta non riteniamo di portare un esempio di applicazione in quanto ormai la stringa di formattazione può contenere sia il " " che il egno "+" a sinistra o a destra, sia il segno "—" a destra elencare tutti i casi possibili servirebbe solo a complicare le idee

— **carattere "_"** - Il carattere "sottolineatura" (in inglese "underscore") ha un effetto molto piacevole, anche se non molto noto in particolare il carattere "—" all'interno della stringa di formattazione fa stampare così com'è il "prossimo carattere" della stringa stessa

Subito un esempio

```
PRINT USING "—(## ##—)" 33 44
```

fa stampare in output

```
(33 44)
```

Lasciamo ancora una volta al lettore la prova di caratteri vari subito dopo "—", per ottenere stampe di effetto

— **carattere "%"** - In realtà non è un carattere da porre nella stringa di formattazione, ma compare "naturalmente" in output allorché il numero da stampare ecceda lo spazio riservato. Appare anche quando l'arrotondamento di un numero che altrimenti non darebbe problemi, genera invece un numero che non "ci sta" nella stringa data

Capita infatti che usando l'istruzione

```
PRINT USING "## ##",555 111
```

si ottiene

```
%555 11
```

Come pure, nel caso in cui si ha

```
PRINT USING " ### , 9999
```

ecco apparire in uscita

```
%1 000
```

Si è visto dunque che la PRINT USING si comporta in alcuni casi molto intelligentemente notevolissimo e l'arrotondamento, che non costringe l'operatore a complesse istruzioni di stringa per avere un numero decimale approssimato Prima di terminare ci sono da aggiungere due importantissime considerazioni

La prima è che vi è un limite al numero di cifre prefissabili con la stringa di formattazione, limite posto a 24, oltre il quale l'MBASIC segnalerà un

```
"ILLEGAL FUNCTION CALL"
```

L'altra considerazione è che tutto quanto detto finora vale anche per l'istruzione LPRINT USING, con la sola differenza che l'output stavolta si avrà sulla stampante e non sul video ed assumendo una lunghezza della linea di stampa pari a 132 caratteri Con questo concludiamo questa puntata ed inoltre formiamo uno specchietto riassuntivo di tutto quanto esposto finora, in forma comprensibilmente succinta. MC

EASY COMPUTING

Ora EASY COMPUTING
ti dà una mano per far funzionare
al meglio il tuo COMMODORE 64.
Una organizzazione amica ed efficace
famosa in Europa, e da oggi anche in Italia.

EASY COMPUTING ti offre la più vasta gamma di prodotti originali per il COMMODORE 64, tradotti in italiano, per un immediato utilizzo, sia nel campo professionale che nel tempo libero. Con il vantaggio di ricevere tutta la documentazione relativa al programma che ti interessa direttamente a casa tua. Basta compilare il coupon o scrivere direttamente a EASY COMPUTING - Via A. Bertani 24 - 50137 Firenze.

Questi i principali programmi che EASY COMPUTING ha selezionato per te:

SUPERSOFT - MUSIC MASTER, BUSICALC 2, BUSICALC 3, TOOLKIT, VICTREE, ZOOM, INTERDICTOR PILOT, MIKRO ASSEMBLER e una scelta di VIDEOGAMES intelligenti.

ABACUS - ZOOM PASCAL, SUPER DISK UTILITIES, SCREEN GRAPHICS, ULTRABASIC, SYNTHY 64, VIDEOBASIC, GRAPHICS DESIGNER, TAS, CADPAK, CHARTPAK.

VIZA - VIZASPELL, VIZAWRITE,

ANIROG - Per la prima volta in Italia decine di videogames originali, considerati come i più elaborati e affascinanti del mercato europeo.
OXFORD PASCAL, HARDCOPY.

HARDWARE - SUPERSKETCH, VIDEO GRAPHIC DIGITISER, LIGHT PEN, 4 SLOT MOTHERBOARD.
INTERFACCE: SERIELINK/RS, SERIELINK, CENTROSERIAL, PRINTLINK, etc.

Sono interessato a ricevere il catalogo generale EASY COMPUTING, gratuitamente e senza impegno, al seguente indirizzo:

Nome ____________________
Cognome ____________________
Indirizzo ____________________
Città ____________________ CAP ________
Professione ____________________
Tel. ____________________

MC

i trucchi del CP/M

di Pierluigi Panunzi

Le funzioni del BDOS

Continuiamo in questo numero lo studio delle varie funzioni del BDOS, studio iniziato due numeri fa e interrotto lo scorso numero da un "ritorno" dell'MBasic.

Prima di proseguire ricordiamo, rimandando per i dettagli al n° 37 di MC, che il "numero della funzione" così come appare nel titolo di ogni paragrafo, è il valore da porre nel registro C prima di effettuare la chiamata alla locazione 5 (CALL 005).

Funzione 3: Read "Reader" Byte

È una funzione particolare del BDOS in quanto consente di leggere un carattere dal dispositivo logico "Reader" (RDR:).

In pratica, bisogna vedere qual è l'effettivo dispositivo fisico "connesso" al CP/M con l'attributo "RDR:", infatti, mentre le prime macchine utilizzanti il CP/M erano dotate di un dispositivo di lettura/scrittura di nastro perforato, per cui aveva senso leggere un byte da tale supporto fisico, oggigiorno nessun personal è dotato di tale arcaico dispositivo di lettura/scrittura. Questo fatto comporta che, a livello BIOS, tale funzione è implementata o meno a seconda delle scelte del costruttore e a seconda di eventuali dispositivi prescelti.

Infatti se il dispositivo "Reader" non è in alcun modo implementato, allora la routine in questione in generale risponde come se avesse letto il byte esadecimale 1AH, usato nel CP/M come indicatore di "End of file".

Viceversa si potrebbe usare il dispositivo logico "Reader" come mezzo con cui utilizzare ad esempio un joystick, oppure un canale di ricezione parallelo ad 8 bit, eccetera, ma il tutto richiede una buona dose di esperienza sia a livello hardware, per la costruzione di un'adatta interfaccia, sia a livello software in quanto questa routine attende, sì, un carattere dal dispositivo, ma fatalmente inchioderebbe tutto il sistema se tali caratteri non arrivassero più: questo perchè la routine "non" aspetta un certo numero di caratteri, "nè" si ferma alla ricezione di un certo carattere.

Funzione 4: Write "Punch" Byte

Questa funzione è analoga alla precedente, solo che tratta l'output di un certo carattere al dispositivo "PUN:"

Anche in questo caso si tratta di una "reliquia" dei primi sistemi su cui operava il CP/M, dotati per l'appunto di un dispositivo perforatore di nastro (Puncher), quale supporto di memoria non volatile.

Nel nostro caso, ancora una volta, il funzionamento della "Write Punch Byte" è legata all'hardware della macchina, ma in maniera meno determinante: in questo caso infatti basta che la routine di gestione di tale dispositivo (a livello BIOS) sia formata dalla sola istruzione RET (Return).

Se si vuole, si può utilizzare il dispositivo "Puncher" ad esempio come dispositivo di uscita (seriale o parallelo) oppure come interfaccia verso un programmatore di E-PROM: anche in questo caso sta al costruttore del sistema o all'utente gestire l'opportuno programma di comunicazione all'interno del BIOS.

Comunque, qualsiasi sia la scelta, il carattere da inviare in output deve essere posto nel registro E, per poi chiamare la routine in questione con il valore 4 nel registro C.

Funzione 5: Write List Byte

Come dice il nome, questa funzione invia il byte, posto nel registro E, al dispositivo "LST:"

Anche se a prima vista non sembrerebbe, dato che praticamente tutti i sistemi dotati di CP/M sono dotati di interfaccia per stampante, anche in questo caso bisogna stare attenti.

Infatti questa funzione non è assolutamente in grado di segnalare al computer lo stato della stampante, come dire che se tale dispositivo non è "On Line" (come capita spesso!), il programma chiamante aspetterà all'infinito, oppure fino a che l'operatore non abbia la curiosità di vedere se la stampante è effettivamente in linea.

Altro problema è dato ad esempio se finisce la carta, ma in genere quasi tutte le stampanti danno un'indicazione ottica (led lampeggiante) o acustica (un sonoro bip) in casi simili.

Anche in questo caso, perciò, il corretto funzionamento della routine di output dipenderà da come è stata implementata la routine a livello BIOS.

Funzione 6: Direct Console I/O

Questa è una funzione che permette l'input o l'output di un byte da o verso la console: è praticamente l'insieme delle due routine di input e di output per il dispositivo "CON:". Vi sono però alcune limitazioni di cui ora parleremo: vediamo come si effettua la chiamata a tale routine.

In questo caso si utilizzano il registro E e l'accumulatore A: nel caso di input di un byte, nel registro E dobbiamo porre il valore ØFFH, mentre in A avremo Ø se non vi è alcun carattere in arrivo oppure un valore diverso da Ø, e cioè proprio il carattere arrivato.

Viceversa, per quanto riguarda l'output, basta che il registro E "non" contenga ØFFH, ma proprio il byte da inviare: in questo caso A non serve.

Ecco dunque quali sono le limitazioni: non si può ricevere il byte nullo (ØØ) nè si può inviare il byte ØFFH, in quanto, come visto, tali valori servono da "segnalatori".

È chiaro che questo comportamento può non essere quello desiderato, ad esempio se desiderassimo ricevere o inviare anche i due byte visti: viceversa, con un'opportuna programmazione dell'IOBYTE (vedere le prossime due funzioni) si potrebbe, ad esempio, "connettere" il dispositivo considerato RDR: alla console per aversi in questo caso il test del dispositivo, ricevendo in A o il byte oppure un valore nullo nel caso in cui tale dispositivo non sia in grado di fornire un byte.

Un break prima di proseguire

Abbiamo parlato dell'IOBYTE e del fatto che le prossime due funzioni lo utilizzano: ebbene, dato che l'argomento è alquanto vasto e delicato, abbiamo pensato di trattarlo per intero nella prossima puntata, soprattutto per non essere costretti ad interromperlo questa volta per i soliti motivi di spazio: saltiamo perciò le routine 7 ed 8 (rispettivamente "Get IOBYTE Setting" e "Set IOBYTE"), rimandando al prossimo numero di MCmicrocomputer, mentre continuiamo la nostra "rassegna".

Funzione 9: Display "$" Terminated String

Questa funzione è molto utile per visualizzare sullo schermo della console dei messaggi, segnalazioni varie o richieste di input di dati via tastiera: infatti tale funzione invia in output al dispositivo "CON:" una stringa di caratteri terminante con il simbolo "$".

Come si vede, tale routine è molto utile, ma appare anche evidente che il messaggio da inviare al video non potrà contenere al suo interno il carattere "$", pena il troncamento del messaggio stesso al sopraggiungere di tale delimitatore.

La chiamata a tale funzione si effettua ponendo nella coppia di registri DE l'indirizzo della stringa "$" - terminated e ponendo in C il valore 9: con la consueta CALL ØØØ5 innescheremo per l'appunto la nostra funzione di output.

Funzione 10: Read Console String

È questa una funzione molto utile in quanto consente l'input dalla console (dalla tastiera, cioè) di una stringa, in genere un comando, terminante come è ovvio con un "Carriage Return" o "Return" o "Enter" che dir si voglia.

La potenza di questa routine risiede nel fatto che consente tutti i possibili controlli

di editing, quali il "Backspace", ecc.

In questo caso l'operatore può infatti "ritornare sui suoi passi" cancellando quanto scritto finora, tramite il Control-U o Control-X, per poi riscrivere una nuova stringa.

Vediamo ora la chiamata a tale funzione: nella coppia di registri DE dovrà essere posto l'indirizzo di memoria del Buffer che conterrà la stringa impostata dall'operatore, nel registro C invece andrà il valore (decimale) 1Ø o meglio (dato che presumibilmente lavoriamo in linguaggio macchina) ØAH, come ormai sappiamo.

Ma vediamo ora più da vicino la struttura di questo Buffer: stabiliamo di avere a disposizione, per il comando da inviare da tastiera (la stringa in input), 128 caratteri (8ØH) dei quali poi presumibilmente useremo solo una parte: al limite uno solo, come pure nessuno.

Allora il nostro buffer sarà così concepito:

— il primo byte indicherà la lunghezza del buffer stesso: nel nostro caso 8ØH, valore che dovremo impostare noi.

— Il secondo byte invece ci indicherà "alla fine" quanti sono i caratteri impostati da tastiera e ci potrà essere utile come contatore nella successiva analisi di quanto digitato.

— I successivi (nel nostro caso 128) byte costituiscono il buffer stesso.

Ovnque come regola bisogna prevedere due byte in più in testa al buffer, byte che verranno gestiti dalla routine stessa del BOOS e che poi utilizzeremo noi alla fine.

Vediamo ora alcuni casi: se ad esempio digitiamo qualcosa, cancelliamo alcune parti, le riscriviamo, ecc, alla fine il secondo byte del Buffer ci darà la lunghezza corretta della stringa impostata, come dire che nel Buffer vedremo l'ultima versione della stringa e nel secondo byte la sua lunghezza.

Attenzione che il carattere "di consenso" e cioè il "Return", NON comparirà nella stringa finale, come dire: se in risposta ad un certo prompt rispondiamo con un semplice "Return", il nostro buffer conterrà tutti zeri ed il secondo byte sarà anche lui nullo, ad indicare che la "stringa in input" era di lunghezza nulla, dal momento che il "Return" non conta. Viceversa però un secondo byte nullo vuol dire che l'operatore ha premuto "Return", cosa che può essere utile in certi casi.

Funzione 11: Read Console Status

Anche questa è una funzione molto utile: ci dice se "in quel momento" c'è un carattere proveniente dalla console ed in attesa di essere elaborato.

È importante notare la differenza con la funzione n. 1 (Read Console Byte), la quale invece aspetta che ci sia un carattere da elaborare, con un'attesa che può diventare molto lunga.

Nel nostro caso invece, la routine testa lo stato della tastiera (o meglio della console) ponendo in accumulatore il valore Ø se non vi è alcun carattere in attesa e viceversa ponendo il valore ØFFH.

Inutile segnalare l'importanza di questa funzione, molto veloce nel senso che non può inchiodare il sistema ed utile laddove l'elaborazione non si deve fermare per attendere un dato da tastiera, ma viceversa deve ad esempio essere fermata dal tutto alla pressione di un certo tasto.

In quest'ultimo caso alla chiamata della routine in questione dovrà seguire il test dell'accumulatore: se è nullo l'elaborazione può proseguire per la sua strada.

Se viceversa il contenuto è ØFFH allora si dovrà procedere all'effettiva lettura del byte con una chiamata alla routine n. 1, che perciò non dovrà attendere la pressione del tasto.

Funzione 12: Get CP/M Number

Ecco una funzioncina semplice semplice, il cui uso è tutto sommato alquanto limitato, solamente nei casi in cui è effettivamente determinante per il corretto proseguimento dell'elaborazione il sapere con quale versione del CP/M stiamo lavoran-

— riporta il valore del DMA Address a ØØ8ØH

— stabilisce lo stato di Read/Write per tutti i dischi.

È una funzione molto utile quando l'operatore deve effettuare il cambio di un dischetto: normalmente infatti, quando l'operatore compie questa (a volte) deleteria operazione, il CP/M non ne sa assolutamente niente, dato che non c'è alcun modo per lui di sapere che lo sportellino del driver è stato aperto! Ecco che dunque per lui il dischetto "non" è stato cambiato. Però il CP/M ha in memoria una tabella opportunamente legata al dischetto vecchio ed in particolare alla situazione di allocazione dei vari settori del disco stesso.

Ora, cambiando tale dischetto, con elevatissima probabilità la nuova situazione di allocazione dei settori sarà completamente diversa da quella memorizzata: ecco che dunque il CP/M, per evitare danni, segnala (con il ben noto quanto odiato messaggio "BDOS ERROR ON A R/O") che non può scrivere sul dischetto, nel caso fosse comandato in tal senso.

Funzioni del BDOS in dettaglio

Valore di C	Nome	Input	Output
3	Reader Input	—	A = carattere ASCII
4	Punch Output	E = carattere ASCII	—
5	List Output	E = carattere ASCII	—
6	Direct Console		
	Input	E = Ø FFH	A = carattere ASCII
	Output	E = carattere ASCII	—
9	Print String($)	DE = indirizzo stringa	—
10	Read Console Buffer	DE = indirizzo buffer	dati nel buffer
11	Get Console Status	—	A = stato
12	Get Version Number	—	HL = numero versione
13	Reset Disk System	—	—

Tabella 1

do. La routine in esame fornirà nella coppia di registri HL un codice ben determinato, dato da:

— H vale Ø se si tratta del CP/M, mentre vale Ø1H se si ha a che fare con l'MP/M.

— L indica il numero della versione:

— ØØH per tutte le versioni precedenti la 2.Ø (ad es. la 1.4)

— 2ØH per la versione 2.Ø

— 21H per la versione 2.1

— 22H per le versione 2.2 e così via.

Come dicevamo l'utilizzazione pratica di tale funzione si ha in un numero ristretto di casi: ad esempio quando si trasportano programmi, che girano in CP/M versione 2.X, in vecchi CP/M (ad esempio il già citato 1.4) non dotati ad esempio di routine di I/O di File Random.

Funzione 13: Reset Disk System

L'ultima routine che analizziamo in questo numero riguarda il reset del sistema a livello dischi.

Non ci sono parametri in ingresso o in uscita e la funzione effettua le seguenti operazioni:

— resetta le tabelle interne del BDOS

— seleziona A: come disco di default

Usando invece la funzione in esame, l'operatore può cambiare il dischetto sotto controllo del programma: ad esempio tale programma potrebbe:

— inviare un messaggio alla console del tipo "Cambiare il disco e premere Return quando si è pronti"

— attendere la pressione del tasto "Return"

— chiamare la routine 13 del BOOS per resettare il sistema dei dischi, per assicurare un corretto funzionamento nel caso di successive scritture del dischetto da parte del CP/M.

Dimenticavamo di ricordare, ma il lettore che è incappato in questa situazione lo sa bene, che nella maggioranza dei casi in risposta al fatidico messaggio di R/O (che sta per Read Only) si può soltanto premere Control-C con conseguenze ovvie: il reset del sistema e la perdita del programma in esecuzione. Concludiamo perciò la puntata con l'ultima delle funzioni generali fra due puntate ci occuperemo delle restanti funzioni, relative alla gestione dei file su disco. Dato che l'argomento è enormemente vasto e alquanto complicato cercheremo di illustrare il tutto con piccoli esempi di programmi.

I prezzi riportati nella Guidacomputer sono comunicati dai distributori dei vari prodotti e si riferiscono alla vendita di singoli pezzi all'utente finale. Sui prezzi indicati possono esserci variazioni dipendenti dal singolo distributore. Per acquisti OEM e comunque vendite multiple sono generalmente previsti sconti qualità. I dati sono aggiornati a circa 20-30 giorni prima della data di uscita in edicola della rivista MCmicrocomputer non si assume responsabilità per eventuali errori o variazioni.

Tutti i prezzi sono IVA esclusa

COMPUTER - PERIFERICHE - ACCESSORI

A C T
Harden Italia S.p.A
Milano Fiori Palazzo T3 Strada N 7
20089 Rozzano (MI)

H00300 F1e 128K RAM, 316K Senza Monitor	2.900.000
H00306 F1 256K RAM, 720K Senza Monitor	3.300.000
H00370 TV Adaptor KIT	100.000
H00373 9" Monitor Monocolore	600.000
H00374 12" Monitor Monocolore	800.000
H00376 Expansion box	700.000
H00377 IR Mouse	300.000
H00310 Portable - 256K RAM 720K	5.500.000
H00315 Portatile - 256K RAM, 720K, Colour/mouse	5.960.000
H00320 Portatile - 512K RAM 720K, Colour/mouse	6.960.000
H00385 I.R. Mouse (nero per portatile)	300.000
H00325 PC 0 256K RAM 315K senza monitor	4.300.000
H00330 PC 1 256K RAM 630K senza monitor	4.500.000
H00335 PC 2 256K RAM, 1440K senza monitor	5.600.000
H00340 Xi 6 256K RAM 5 Mb senza monitor	7.400.000
H00345 Xi 10 256K RAM, 10 Mb senza monitor	8.600.000
H00346 Pacch Xi 10 (12") + MT 180 L + H. azienda	15.000.000
H00350 P 7 512K RAM, 10 Mb senza monitor	12.500.000
H00390 Mouse per PC, Xi, P.7	500.000
K00392 Co processore matematico 8087	790.000
K00394 Scheda colore	1.100.000
K00396 H R 8" Monitor - mono col (G o B)	600.000
H00397 H R 12" Monitor mono col (B o G)	800.000
H00410 Set di 10 cavi 3 Mt. per P 7	90.000
H00411 Set di 10 cavi 8 Mt per P 7	185.000
H00412 Set di 10 cavi 15 Mt per P 7	260.000
H0035h Point 32 256K RAM 10 Mb File server	10.000.000
H00380 Point 32 256K RAM 20 Mb File server	14.600.000
H00420 Unità di Back - up con cartuccia 100 Mb	6.000.000
H00421 Network T - box	600.000
H00188 Set borsa apricot	70.000
H00450 10" Monitor colore	1.195.000
H00452 Unità esterna 10 Mb	4.200.000
H00454 Scheda espansione 128K	740.000
H00455 Scheda espansione 256K	1.480.000
H00456 Scheda espansione 512K	2.960.000
K00460 Modem interno	1.000.000
H00465 Conf 10 dischetti S.S	120.000
H00007 Stampante MT 180V	2.050.000
H00008 Stampante MT 180L	1.950.000
H00024 Stampante MT 80A	760.000
H00031 Stampante QUME 10/35	3.860.000
H00032 Stampante MT 440 L CTX	4.200.000
H00033 Stampante MT 440 L Seriale	4.300.000

ADDS
Transpart SpA
Corso Sempione 75 - 20145 Milano

Viewpoint VP A 1 Video terminale 12" 25 righe×80 caratteri matrice 5×8 tastiera separata e tastierino numerico	1.377.750
Viewpoint VP A 2 Unità come il VP A 1 con indirizzamento indipendente del cursore e abil./disab. della porta ausiliaria	1.377.750
Viewpoint VP 3A + Unità compatibile con terminale video LEAR SIEGLER ADM 3	1.377.750
Viewpoint VP 60 Unità base come VP A 1 con caratteristiche di display (matr 7×8), configurazione di tastiera, modi operativi, editing	1.881.000
Viewpoint VP 60 Unità base come VP 60 con riga di stato programmabile, 15/30 funzioni programmabili 256 caratteri o simboli definibili	2.807.000
Viewpoint COLOR Unità base paragonabile al VP 60 con matrice 5×6 e con 8 colori in foreground e 8 colori in background	3.234.000
Viewpoint VP 78 Unità completamente compatibile con il terminale IBM 3278-2	2.805.000
Viewpoint VP 825 (EPICS)	1.881.000
Viewpoint VP G Unità base con capacità grafiche	4.430.250
Viewpoint VP 78 a colori	4.240.500
Viewpoint VP 80 G Unità con grafica evoluta	5.032.500

Nota. prezzi per dollaro a L. 1.900

ADVANCE (U S A)
Condor Informatics Italia
Via Gran Sasso 8 20145 Milano

Advance 86 A 8086 128K RAM	1.560.000
Expansion Box - 2 semifloppy da 320 K per Advance 86 A	2.760.000
Advance 86 B 86 A + Expansion Box + Monitor + 4 Pacchetti Applicativi	4.660.000
Advance Mod B 128K RAM colore grafica standard 10 Mb floppy da 360K. Hard disk da 10Mb 4 pacchetti applicativi-monitor	7.420.000

Nota prezzi per 1$ 1000 lire

ALPHACOM (U S A)
Reba Computer G.B.C. Italiana S.p.A.
Viale Matteotti, 66 20092 Cinisello Balsamo (Milano)

Stampante ALPHACOM 32 per collegamento a Computer Sinclair	236.000

ALTOS (U S A)
Amtele
Via Tonale, 12 20124 Milano

ACS 8000/10 208K RAM 1 floppy 500K + 1 HARD DISK 8" incorporato 10Mb	18.530.000
ACS 8000/12 208K RAM 1 floppy disk da 500K + 1 HARD DISK 8" da 20 Mb	20.230.000
ACS 8000/14 208K RAM 1 floppy disk 500K + hard disk 8" da 40 Mb	24.650.000
ACS 580/2	6.120.000
ACS 580/20 192Kb RAM + 1 floppy da 1 Mb + 1 winchester da 22 Mb	10.370.890
ACS 580/40	12.750.000
ACS 586/20	21.250.000
ACS 586/40 512K RAM 1 floppy da 1 Mb + 1 winchester da 40 Mb	25.500.000
ACS 68000/12 16 bit 500Kb + 1 floppy da 500K con microprocessore Motorola 68000 + 1 hard disk da 20 Mb	30.260.000
ACS 68000/14 16 bit 500Kb + 1 floppy da 500K + 1 hard disk da 40 Mb con microprocessore Motorola 68000	34.850.000
ACS 8600/12 1 H D da 20 Mb 512Kb	26.350.000
ACS 8600/14 1 H D da 40 Mb - 512Kb	29.750.000
ACS 986/40 1024Kb 1 H D da 42Mb	30.430.000
U/K 10/580 winchester addizionale per 580/10 e 580/20 da 12 Mb	1.090.000
MTU 2 unità a cassetta magnetica da 17 Mb per back up disco rigido	6.630.000
MTU 3 unità a cassetta magnetica da 17 Mb per back up disco rigido	6.630.000
MTU 6 unità a cassetta magnetica da 17 Mb per back up disco rigido	6.630.000
RAM U/K espansione di memoria centrale a 1 Mb	4.590.000

Software di base:	
CP/M	340.000
MP/M-86	11.050.000
BASIS	1 445 000
XENIX	1.091 500
UNIX	1 801 500
BM/COS	2.550 000
1 $ - 1700 lire	

AMSTRAD (G.B.)
DEI
L.go Porta Nuova, 14 24100 Bergamo

Amstrad CPC 464 con monitor a fosfori verdi	739 000
Amstrad CPC 464 con monitor a colori	1 030 000
Stampante con cavo	809 000
Controller dischi + 1° drive	849 000
2° drive	519.000
Joystick	29 000
Manuale d'uso in italiano	24 000
Software professionali	da 20 000 a 140 000

ANADEX INC. (U.S.A.)
Transpart S.p.A.
Corso Sempione, 75 20125 Milano

DP 8000B Stampante	2.048.250
DP 9500B Stampante	2.821 500
DP 9620B Stampante	3.036 000
DP 9625B Stampante	3.298 350
DP 9725 B Stampante	3 572.250
WP 6000 Stampante	8 898.780
DP 6500 Stampante	8.542.250
Nota: prezzi per dollaro a L. 1 800	

APPLE COMPUTER (U.S.A.)
Apple Computer Italia
Via Bovio, 5 42100 Reggio Emilia

Apple IIe 64 K RAM	1 309.350
Scheda 80 colonne	225.000
Scheda 80 colonne con espansione a 128 K	485.000
Monitor IIe	365 000
Disk II, drive e doppio controller	700 000
Disk II, drive aggiuntivo	600.000
DuoDisk 2×140 K	1 150.000
ProFile 10 megabyte	4 100.000
Hand controller IIe/IIc	80 000
Joystick IIe/IIc	106 000
Numeric Keypad IIe	298 000
Tavoletta grafica per IIe	1 490 000
Mouse per IIe	270.000
Interfaccia seriale per IIe	250.000
Interfaccia parallela per IIe	300 000
Interfaccia IEEE 488 per IIe	841 000
Scheda prototyping hobby per IIe	43 000
Apple IIc 128 K RAM 1 minifloppy integrato	2.399 350
Monitor IIc	385 000
Flat Panel Display	1.290 000
Supporto per monitor IIc	72.000
Disk IIc aggiuntivo 140 K	588 000
Mouse per IIc	180 000
Borsa per IIc	75 000
Macintosh 128 K RAM - video - 1 microfloppy integrato da 400 K mouse Paint /Write	4 496.350
Macintosh 512 K RAM video 1 minifloppy integrato da 400 K mouse Paint/Write	5.990 350
Unità microfloppy aggiuntiva 400 K	880 000
Numeric Keypad	180 000
Macintosh XL 1 microfloppy + ProFile 10 MB	9.999 000
Stampanti	
Image Writer 80 col. 180 cps grafica	1 100 000
Image Writer Stand	118 000
Image Writer 132 col	1 468 000
Stampante a margherita	3.636 000
Scriba 7 colori a trasferimento termico	549 000
Plotter (interf. seriale)	1.850.000

Accessori e periferiche non di produzione Apple Computer

Scheda orologio calendario CCS	280 000
Modulatore UHF	82 000
Interfaccia CCS seriale RS232-C	347 000

AQUARIUS
Aequo s.r.l.
Via S. Gallo, 16/R 50129 Firenze

Computer Aquarius 1 4K RAM	199 000
Aquarius printer	274 000
Data Recorder	140 000
4K RAM Expansion	43 000
16K RAM Expansion	100 000
32K RAM Expansion	172.000
Printer	239 000
Miniexpander (include anche due joystick)	144 000
Cartucce giochi	da 40.000 a 87 000
Cartucce Fileform - Finform Ext. Basic	172 000
Cartucce Logo Aquarius	172.000

BARCO
TELAV International
Via L. Da Vinci, 43 20090 Trezzano S.N. (MI)

Monitor 16" DCD 1640	1.071 000
Monitor 22" DCD 2240 a colori	1.257 000
Monitor 27" DCD 2740 a colori	1 367 000

BASF
Data Base Sistemi srl
Via Don Griffanti, 2 21047 Saronno (VA)

7105 - 48 K RAM macchina slave	8 000 800
7120 64 K RAM, 3 minifloppy da 163 KB	8 000 000
7125 68 K RAM 3 minifloppy da 320 KB	8 900 800
7130 80 K RAM disco 5 MB+minifloppy 400 KB	11 400 000
7161 Modulo aggiuntivo disco 5 MB 5"	3.800 000
Interfaccia seriale RS232 aggiuntiva	750 000
6104B floppy disk drive 8" doppia faccia	1 809 660
61055L floppy disk drive 8" doppia faccia Slim	1.025 480
6128 floppy disk drive 48 TPI doppia faccia Slim	443.920
6138 floppy disk drive 5,25" doppia faccia Slim	532 580
6185 Drive 5.25" Winchester	2.048 480
6180 5,25 Winchester 10Mb Slim	1 080 700
6238 Dual floppy disk drive doppio 5,25"	1.036.640

CALCOMP (U.S.A.)
Calcomp S.p.A.
Palazzo F1-20090 Milanofiori Assago (MI)

Plotter M80 (8 penne, A4)	2 850 000
Plotter 81 (8 penne, A3)	8 950 000
Plotter 1082 (Dual-mode A0)	20 300 000
Plotter 1043 (Foglio singolo A0)	20 300 000
Plotter 1048 (Dual-mode, A0)	28.000.000
1 $ = 2000 lire	

CANON
Canon Italia S.p.A. - Via dell'Industria 13 37012 Bussolengo (VR)

Canon X07 portatile con display inc. programmi in basic microsoft da 8K RAM a 24KRAM da 20KROM a 40KROM programmi su schede Interf. RS 232 parallela Comp. Centronics + 1 plotter 4 colori	620.000
Stampante per X-07	446.600
AS 100/M	4.825.000
AS 100/C	8.625.000
A-1111 Tastiera	325 000
A-1300 2×5" floppy disk drive	2.850.000

A-1330 2×8" floppy disk drive	5 700 000
A-1350 10 Mb Hard disk + 8" FD	7.500 000
A-1200 Stampante grafica	1 490 000
A-1210 Stampante colori ink-jet	1 800 000
A-1284 MATRIX PRINTER 184 Colonne	2.280 000
A-1011 I/O Extension/board	250.000
A-1500 V24/RS232C	260 000
A-1502 Centronics Interface	250.000
A-1503 Communicato Interface	280.000
A-1020 128 Kb RAM	875.000
A-1021 256 Kb RAM	890 000
A-1022 384 Kb RAM	1.305 000
A 1001 RAM Chip 128K	460 000
A-1002 Clock Set	160.000
A 1004 32 Kb VIDEO RAM	250.000
A 1100 Pointing Device	125.000
JI-20 Cassetta colore per A 1210	33.000
JI 200 Cassetta inchiostro nera per A 1210	10 000

CASIO (Giappone)
Ditron S.p.A. – V.le Certosa, 138 – 20156 Milano

FP-1000 Unità centrale B/N	1 129 000
FP 1100 Unità centrale colore	1.311 000
FP-1001 Monitor verde	422 000
FP-1004 Monitor colore	1.263 000
FP 1020 Disk drive da 640 Kbyte	2.187 000
FP 10282 Disk Drive da 1Mb	3 700 000
FP-1030 Espansione RAM 16K	405 000
FP-1031 Contenitore ROM fino a 16K	73 000
FP 1060 Espansore 4 porte I/O	874 000
FP-2000 Sistema Operativo CP/M 2,2	182.000
RX 80 Stampante 100 cps 80 col. grafica	960 000
FX-80 Stampante 160 cps. 80 col. grafica	1 400 000
MX 100 Stampante 100 cps 132 col. grafica	1.680 000
FP-200 – computer portatile 8K RAM	675 000
AD 41005 – adattatore corrente	48 000
EP 201 – espansione 8K RAM	118 000
FP-1035 – interfaccia RS 232C	129.000
FP 1083 – cavo per RS 232C	69 000
FP 1021 – disk drive da 70 KB	874 000
FP 1011 – stampante – plotter 4 col.	440 000
FP-1085 – cavo per stampante – plotter	84 000

CAT
Telcom s.r.l. — Via Matteo Civitali, 75 - 20148 Milano

Accoppiatore acustico	800.000
Nota: prezzo per dollaro a L. 2.000	

CENTRONICS DATA COMPUTER CORP (U S A)
Centronics Data Computer Italia S.p.A.
Via Achille Grandi, 10 - 20093 Cologno Monzese (MI)

GLP stampante 50 cps.	600 000
H 80	1 100 004
H 136	1.550 000
H RS232	170 000
351	3.000 000
353	4.200 000
353 ASF (1 caricatore)	5.750 000
Caricatore addizionale	960 000
357	6.000 000
358	7.000 000
358 ASF (1 caricatore)	8.500 000
LW 400	11 000 000
LW 800	17 000 000
6085	16 000 000
MDB II	31 000 000
MDB III	38 000 000

CHALKBOARD
Arcona s.r.l.
Via Filippo Lippi, 18 - 20131 Milano

Power Pad – Tavoletta grafica per Apple – Commodore 64 – Atari	320.000
Power Pad – Tavoletta grafica per Commodore 64 – Atari	300.000

COINTREAU (Hong Kong)
JETSET
Via di Torrevecchia 3/F – 00168 Roma

Modem phone modem con telefono integrato per tutti i computer su normali linee telefoniche – velocità 300/1200 Baud	398 000
Joy Stick professionale per Apple	120 000
Joy Stick per Commodore auto fire 5 pulsanti	30 000
Mouse grafico	140 000
Registratore per Commodore	40 000
Scheda Madre per Apple	450.000
ADLIB II	780.000
Scheda scientifica ed industriale 2 canali 12 Bit – Convertitore da digitale ad analogico con out put simultaneo	1 020.000
Scheda scientifica ed industriale – Sistema di acquisizioni dati veloce 8 Canali di input, differenziati – Auto-ranging e fast logging software	1 050 000

COMMODORE (U S A)
Commodore Italiana srl
Via F.lli Gracchi, 48 – 20092 Cinisello Balsamo (MI)

VIC-20	199 000
1210 espansione 3 K RAM	40 000
1110 espansione 8 K RAM	75 000
1111 espansione 16 K RAM	125.000
1211 M 3 K Super Exp.	75 000
1212 Programmers Aid	47.500
1213 Mach. Language Monitor	47.500
1214 Sintetizzatore vocale	175 000
4011 VIC Rel (per controllo Relè)	95 000
4012 VIC Switch (per collegare fino a 16 VIC)	225 000
2011 VIC STAT cartuccia	95.000
2012 VIC GRAPH cartuccia	99 000
T1 Interfaccia IEEE 488 per VIC	175.000
T3 Interfaccia Centronics per VIC	118.000
Commodore 16 16KB RAM	199 000
CBM/64 CPU 64K RAM	625.000
64 Executive (sistema a valigetta con video 5" e minifloppy)	2.350 000
Cartuccia e sistema operativo CP/M per 64	125 000
T2 Interfaccia IEEE 488 per 64	175 000
Commodore Plus 4 64K RAM	645 000
Commodore 8296 128K RAM	1 995 000
1530 registratore a cassette	120 000
1541 minifloppy 170 K	630 000
1601 Monitor monocromatico 12"	285 000
1702 Monitor a colori 14" con audio	690 000
MPS 801 Stampante 8450 Cps	515 000
MPS 802 Stampante 80 Cps	645 000
MPS 803 Stampante bidirezionale 60 cps 80 col.	595 000
MCS 801 Stampante ad aghi a colori	905 000
DPS 1101 Stampante a margherita	1 195 000
1520 Stampante plotter a 4 colori	375 000
1311 Joy Stick per Vic e 64	13 500
1312 Paddle per Vic e 64	22.500

COMPITANT
Compitant – Via Vittorio Emanuele II, 9 – 91021 Campobello di Mazara (TP)

Compucolor IICP/M 280 grafici colore 570 per 192 compatibilità APPLE completo monitor RGB con floppy 5" da 152K	4 000 000
Floppy aggiuntivo	600.000
Compucolor III con microprocessore 80/60 grafici colore 128×128 completo di monitor RGB con 1 floppy 8" 102K	3.800 000
Floppy 8" aggiuntivo	1 900 000
Mod. Leonard monitor 14" 8 colori + drive 8" + stampante	6.980 000
Compucolor Executive 16K con floppy 92K	5.818 000

COMPUTER COMPANY
Computer Company s.a.s.
Via San Giacomo 32 – 80133 Napoli Tel. 081/310487 324786

TIN 100 64 K RAM 1 MB	11 000 000
TIN 200 64 K RAM 2 MB	12.300 000
TIN 804 64 K RAM 4 MB	15 900 000
TIN 810 64 K RAM (10Mb+1Mb)	18 500 000
TIN 820 64 K RAM (20Mb+1Mb)	22.150 000
TIN 830 64 K RAM (30Mb+Mb)	25 560 000
Unità a floppy disk 1 Mb	2.750 000
" " " 2 Mb	3 200 000
Computer TIN 64 K (terminale intelligente)	0 480 000
Scrivente	495 000
Scheda espansione per TIN 64 K	950 000

CORVUS SYSTEMS (U S A)
Citra Dinci s r l. – Casella Postale, 58 – 40062 Zola Predosa (BO) Tel 051/752237 8 linee

OMNIDRIVE (Unità a disco con integrato il Disk Server per la rete Locale OMNINET)	
— 5 5 MB formattati	5.270 000
— 11 1 MB formattati	6 750 000
— 16 8 MB formattati	0 490 000
— 45 1 MB formattati	13 270 000
TRANSPORTER	1.319 000
per	
IBM PC/XT APPLE II IIE, III, DIGITAL RAINBOW 100 TEXAS TI PROF ZENITH Z 100	
BANK	6 290 000
Unità a nastro removibile da 100 a 200 MB ad accesso anche RANDOM per	
OMNINET MIRROR	2 837.000
Unità di Backup su video cassette (da acquistare a parte)	
MIRROR PER IBM XT	1.319 000
Unità di Backup per IBM XT su video cassette (da acquistare a parte)	
OMNISHARE	2.637 000
Rete OMNINET per IBM XT	
CORVUS CONCEPT	
Workstation da 256KB RAM	11 838 000
Workstation da 512KB RAM	13 780 000

COSMIC (Italia)
Cosmic s.r.l.
Largo Luigi Antonelli, 2 – 00145 Roma

Rack Quasar 1/2 1 drive 8" doppia faccia per Apple	2.870 000
Rack Quasar 2/1 2 drive 8" singola faccia per Apple	2.800 000
Rack Quasar 2/2 2 drive 8" doppia faccia per Apple	3 380 000
GALAXY serie 70	
Mod 72 64K RAM + video 24×80 + 2 floppy S D per 1 Mb	6.550 000
GALAXY serie 80	
Mod 82 64K RAM + video 24×80 + 2 floppy D D per 1Mb	7 050 000
Mod. 82/M2 128K RAM comp MP/M per 2 utenti	8 750 000
GALAXY serie 380	
Mod. 382 64K RAM + video 24×80 + 1 disco winch. da 8 Mb + 1 floppy da 1 Mb	10.300 000
Mod. 382/M2 128K RAM Comp MP/M per 2 utenti	11 100 000
Mod 382/M4 256 RAM Comp MP/M per 4 utenti	11 400 000
GALAXY serie 480	
Mod. 480 64K RAM + video 24×80 + 1 disco winch. da 10 Mb + 1 floppy da 1 Mb	10 900.000
GALAXY serie 680	
Mod. 682 + 1 Floppy da 1 MB + 64K Ram + Video 24×80 + disco da 20 MB	12 260 000
Terminale aggiuntivo per Mod./M2 M4	1 800.000
Hard Disk PHOENIX	
PHOENIX 5 5Mb 3 sistemi operativi (Dos CP/M Pascal) software e interfaccia per APPLE e compatibili	2 500 000
PHOENIX 10 Come Mod 5 con 10Mb	3 400 000
PHOENIX 10/PC Come Mod 10 per IBM Personal Computer	3 700 000
PHOENIX 15 come 5 con 15Mb	4 100 000

CROMEMCO (U S A)
Co.N.I.A. Via A. di Vincenzo 88/E, 40129 Bologna

C 10 64K RAM 28 K ROM video	ǥ2 270 800
C 10SP C 10 con 1 minifloppy da 300K	4 075 500
CKBA tastiera ASCII	446.500
CKBC tastiera ASCII estesa	802.500
CFO minifloppy 390K aggiuntivo	1.358 500
CS 1 Z80 64K RAM 2 floppy	8 407 000
CS 102 MC68000 288K RAM 2 floppy 8.538	10.518 500
CS-1HZ2 Z80 256K RAM 1 floppy + hard disk 21M	18 988 500
CS 1H05 MC6800 288K RAM 1 floppy + hard disk 21M	17.879 000
CS 2 Z80 64K RAM 2 floppy	9.860 900
CS 202 MC68000 256K 2 floppy	12.825.500
CS 2H Z80 64K 2 floppy + hd 21M	17 979 000
CS 2H02 MC68000 2 floppy + hd 21M	19.988 000
CS 3A Z80 64K RAM 2 floppy 1M	14 725 000
CS 302 MC68000 + 256K RAM 2 floppy 1M	16.334 000
CS 3H Z80 64K RAM 1 floppy + hd 21M	19 948 000
CS 3H02 MC68000 256K RAM 1 floppy + hd 21M	22.097 000
C 5C Terminale 2000 caratteri	2.817.500
Nota prezzi per 1$=1 900 lire	

DATA SOUTH (U S A)
Segi
Via Timavo, 12 20124 Milano

D.S 180 Stampante seriale 180 cps grafica buffer 2K	3 800 000
D S 220 220 CPS trascinamento sotto frizione e trattore	4 826 000
Nota: 1 $ 1900 lire.	

D D P (Italia)
D D P s.r.l DIGITAL DATA PRODUCTS
Corso Moncalieri, 72 10132 Torino

D700 Sottosistema per winchester 6 38 Mb per Apple II	4 500 000
MXP Multiplexer fino a 4 APPLE	1 200 000
D705 Sottosistema winchester 6 38 Mb DEC QBUS RT11 e RSX11M	4 000 000
D710 Sottosistema winchester 12.76 Mb DEC QBUS RT11 e RSX11M	5 300 000
D810 Drive aggiuntivo 12 76 Mb per D710	3 000 000
Nota prezzi legati ad un cambio 1$ = 1800±2%	

DIABLO SYSTEM INC (U S A)
Segi
Via Timavo, 12 20124 Milano

630 API stampante a margherita 40 CPS	6 175 000
630 ECS/PC stamp. a margherita 40 CPS con interf per IBM Personal Computer	7 505 000
630 ECS stamp. a margherita 40 CPS 400 caratteri in combinazione-buffer 1344 byte	7 830 000
Nota 1 $ lire 1980	

DIGITAL EQUIPMENT
Digital Equipment S.p.A.
V.le Fulvio Testi pag V Gorki 105 20092 Cinisello Balsamo (MI)

RAINBOW PC 100	
128 Kb RAM Minidisco duale 2×400 Kb. Monitor Monocr 80/132 CL + tastiera CP/M + MS DOS	8 866 100
RAINBOW PC100 PLUS	
Stesso modello precedente + Winchester 10 Mb	11 835 100
PROFESSIONAL 350	
512 Kb RAM minidisco duale 2×400 Kb, Monitor Monocr 80/132 CL + tastiera, scheda grafica Floating Point, P/OS Multi-tasking	11.218 000
PROFESSIONAL 350 + W	
Stesso modello precedente + Winchester 10 Mb	16 315 000
DECMATE III package	6.549 000
LQP 20/5B come LQP25/AI ma con vassoio doppio	4.320 000
OPZIONI	
Monitor a colori 13"	1 700 000
Scheda grafica	830 700
Supporto per install. verticale	164 000
Espans. memoria 128 Kb	828 900
Espans memoria 256 Kb	1.682 000
Stampante a 100/CPS	1 470 000
Stampante a 240/CPS	3.653 000
Stampante a margherita 32/CPS	6.412 000
Stampante a margherita LQP20/AI 35/CPS	2 880 000
LQP25/5A come LQP/25AI ma con alimentatore fogli vassoio singolo per DECMATE III	3 980 000

DRAGON

ECO s.r.l. — Via Pratosauto, 18 - 37126 Verona

DRAGON 32	450.000
DRAGON 64	500 000
Controller per 4 disk drive	140 000
Cabinet con 1 disk drive	586 000
Cabinet con 2 disk drive	888 000
Coppia joystick	48 000
Cavo per stampante parallela Centronics	20 000

DYNALOGIC Info-Tech Corporation

Transport SpA
Corso Sempione, 75 20145 Milano

Hyperion 256 KB RAM 2 minifloppy da 655 KB tastiera separata monitor 7" (640×250 punti), interfaccia seriale RS 232C e parallela sistema operativo MS/DOS con EDLIN BASIC ed ASSEMBLER	6.785.000
Nota. 1 $ lire 1650	

DYNEER

TECHNITRON Milanofiori Pal. E/2 20094 Assago (MI)

DW16 16CPS 100 Colonne Parallela	1.230 000
DW16 16CPS 100 Colonne - Seriale	1.380 000
DW36 36CPS 132 Colonne - Parallela	2 825 000
DW36 36CPS 132 Colonne - Seriale	3 100 000
DW20 20CPS 132 Colonne Parallela	2 010 000
DW20 20CPS 132 Colonne Seriale	2.135 000
Monitor 12 MKC monocromatico composito	380 000
Monitor 12 MHI monocromatico-uscita rgb	410 000
Monitor 14 CMi colore	1 200.000

EDICONSULT srl

EDICONSULT
Via Rosmini, 3 20052 - Monza

EDI 1000 128Kb RAM - 2 floppy 360Kb int seriale e parallela video monitor monocromatico e tastiera Comp IBM CPU 8088	4.200 000
EDI 1000 con 1 floppy e 1 disco 10Mb	8 300 000
EDI 2000 personal computer CPU Z80 164Kb RAM 2 floppy e 700 Kb cad + int. parallela e seriale video m. tastiera	3 800 000
EDI 2000 con 1 floppy 700K e 1 disco 10Mb	5.800 000
Nota. $ - 2000 lire	

ELETTRONICA EMILIANA s.n.c

Viale delle Nazioni, 84
41100 Modena

Alfetta 16 Base alimentazione + 5Vcc	273 000
Alfetta 16 Panel alimentazione + 8 Vcc	294 000
Alfetta DIN/P (parallela) alimentazione dalla rete	488 000
Alfetta DIN/S (seriale) alimentazione dalla rete	482 000
Alfetta DIN/BCD alimentazione + 8 Vcc	414 000
Scriba 20 Serie di stampanti ad impatto a 28 e 35 colonne con avvolgitore interno, supporto rotolo serratura a chiave, alimentazione dalla rete	
Scriba 21/P per carta in rotolo due colori	1 168 000
Scriba 21S/P modulo continuo con trascinamento sprocket, due colori	1.305 000
Scriba 21V/P per carta in rotolo e validazione su modulo discreto	1 373 000
Scriba 21E/P etichettatrice con spellicolatore automatico	1.505 000
Scriba 22/P carrie su due rotoli scontrino + foglio cassa (per scontrini fiscali)	1 424 000
Scriba 24/P per moduli discreti 5 copie senza limitazione di formato	1.324 000

ENTER COMPUTER, Inc

D.D.P. s.r.l. DIGITAL DATA PRODUCTS
Corso Moncalieri, 72 - 10132 Torino

SWEET P PLOTTER A 4 15cm/sec	980 000
BPS/IBM s/o grafico	730 000
PFS/IBM s/o grafico	450 000
BPS/APPLE s/o grafico	350 000
PRIME/APPLE s/o grafico	850.000
Nota. Prezzi legati ad un cambio 1$ - Lit. 1500±2%	

EPSON (Giappone)

Segi
Via Timavo, 12 - 20124 Milano

HX 20 personal computer portatile con valigetta	1.240 000
LQ 1500 P 132 Col 200 CPS matrice 17×24 con Interfaccia parallela	2 750 000
LQ 1500 S Interfaccia Seriale	2 800 000
Inseritore automatico di fogli singolo e singola vaschetta	880 000
P40 termica 40 col (80 in stampa compressa) 45 CPS seriale e parallela	340 000
JX 80 grafica come FT 80 con nastro a 4 colori e combinazioni fino a 7 colori	2 100 000
QX 10 personal computer CP/M 192 Byte RAM	5.100 000
Microcassetta per Hx 20	280 000
Unità di espansione di memoria 16K per Hx 20	278 000
Espansione ROM a cartuccia per Hx 20	105 000
Lettore di codici a barre per Hx 20	280 000
Cavo per registratore esterno	20 000
Cavo RS 232	50 000
INTERFACCE	
Seriale RS 232C 8143	90 000
Seriale RS 232C con 2K buffer e protocollo X ON X OFF 8148	210 000
IEEE 488 con buffer 2K 8165	160 000
APPLE II per stampa grafica e l'automatico 8132W con cavo	250 000
Parallela con buffer di 2KB 8171	210 000
STAMPANTI	
RX80 80 col 100 CPS	780 000
RX80 F/T 80 col. 100 CPS	870 000
FX100 132 col 160 CPS	1 870 000
FX80 80 col 160 CPS	1 330 000
RX 100 132 Col 100CPS	1 250 000
PX 8 Computer portatile 64Kb RAM 32 Kb ROM CP/M con schermo a 80 Col. per 8 righe microcassetta incorporata	1 970 000
PX 8 versione software integrato	2.250 000
R Disk 120 est Ram disk 120 Kb	680 000

ESPRIT SYSTEM

Segi
Via Timavo 12 - 20124 Milano

Mod Esprit II terminali video con fosfori verdi e tastiera separata	1 520 000
Mod Esprit III terminali video con fosfori verdi con tastiera separata	1 995 000
Mod Esprit III Colore terminali video a 8 colori con tastiera separata	3 040 000
Mod Executive 10 terminali video a fosfori verdi con tastiera separata	2 470 000
Mod ESP 6110	1 501 000
Mod ESP 6310 come Esprit III con video 14"	1 995 000
Mod Executive 10/25 come Executive 10 video 14" fino a 132 col	3 040 000
Mod Executive 10/102 Terminal Video 14" compatibile con la serie DEC VT100	2.600 000
Nota prezzo per dollaro a L. 1 800	

GENIUS COMPUTER s.r.l.

Genius Computer s.r.l.
Via G Carne Pellegrini 24 25100 Brescia

GC 3000 - Z80A a 4MHz - 80KB - 2 minifloppy 1 8 MB formattati - Video - Tastiera alfanumerica separata	8 300 000
GC 3001 Z80A a 4MHz 128 KB - 1 minifloppy da circa 800KB formattati 1 disco fisso 5 MB formattati video tastiera alfanumerica separata Interfaccia seriale RS232C e RS422	11 770 000
GC 3002 Come mod GC 3001 ma con disco fisso da circa 10MB formattati	12 830 000
GC 3003 Come mod GC 3001 ma con disco fisso da circa 15MB formattati	13 540 000
GC 3004 - Come mod. GC 3001 ma con disco fisso da circa 20MB formattati	14 410 000
GC 3000/T 128 KB - Interfacce seriali RS232C e RS422 senza minifloppy da collegare in rete	4 900 000

GNT (Danimarca)

Telcom s.r.l.
Via Matteo Civitali, 75 - 20148 Milano

Mod 3601/50 (perforatore di banda, interfaccia parallela)	2.350 000
Mod 3608/75 perforatore di banda telex, interfaccia seriale e parallela con convertitore ASCII e BAUDOT 75 cps	3.400 000

GRAPHTEC (ex Watanabe)
SPH computer Srl
Via Giacosa, 3 20127 Milano

MP1000-21 MIPLOT 8 penne, 15 cm/sec int. 8 bit parallelo	1 906 000
MP1000-01 MIPLOT 8 penne 15 cm/sec int. RS232 C	2.005 000
MP1000-11 MIPLOT 8 penne 15 cm/sec int. IEEE-488	2.117 000
MP1000-31 MIPLOT 8 penne 15 cm/sec int. 8 bit per DX	2.421 000
DT 1000 digitalizzatore formato A3	1.889 000
WX 4731 plotter a tamburo 4 penne	3 674 000
PC 2621 interfaccia parallela 8 bit	5 001 000
FP 5301 01 Personal plotter 10 penne formato A3-49 CP/M + RS232C.	8 847 000
FP 5301 11 come 5301 01+IEEE 488	6 781 000
FP 5301-21 – come 5301-01 + Interfaccia 8 Bit parallelo	6 305 000
PC 2601 – interfaccia RS 232C	869 000
PC 2611 interfaccia HP IB IEEE488	1 103 000
PC 2602 interfaccia 20 m A Current Loop	1 169 000

Nota. prezzi per 1 Yen – 7,3 lire

GTCO
Talcom srl Via M Civitali 75 20148 Milano

Digitizer GTCO mod DP5 11×11 (tavoletta grafica)	3 080 000
Digitizer GTCO mod MD7 formato 6×6 con stilo	1 000 000
Digitizer GTCO mod DP5 formato 6×6 con cursore	2.150 000
Digitizer GTCO mod MD7 formato 12×12 con stilo	2.430 000
Digitizer GTCO mod MD7 formato 12×12 con cursore	2.650 000
Software per IBM PC	435 000
Alimentatore 12V 275 mA	250.000
Cavo per collegamento	68 000

Nota: prezzo per dollaro a L. 2000

HEWLETT PACKARD (U S A.)
Hewlett Packard Italiana
Via G. Di Vittorio 9
20063 Cernusco sul Naviglio (MI)

Personal Computer HP 85B	8 715 000
Personal Computer HP 86B	4.337 000
Personal Computer Portatile HP 110	7.802 000
Personal Computer HP 150B 45610BZ	8 652 000
HP 150B con dischi 3.5" 9122D 45650BZ	8 923 000
HP 150B con disco 14.5M 91330 – 45660BZ	12 150 000
Stampante termica per HP 150 2674A	773 000
Personal Computer Tecnico 9816S	11.160 000
Scheda di memoria 1Mbyte opz 001	7 108 000
Unità Periferiche	
Dischi flessibili da 3 pollici	
Master doppio 1.4M 9122D	3 178 000
Master singolo 710K 9122S	2.409 000
Master singolo portatile in HP-IL 710K – 9114A	2 018 000
Master doppio 540K 9121D	2.584 000
Master singolo 270K 9121S	1.827 000
Dischi flessibili da 5 pollici	
Master doppio 540K 82901M	5 663 000
Master singolo 270K 82902M	3.882 000
Master singolo da 5" compatibile IBM PC 9125S	3 040 000
Dischi flessibili da 8 pollici	
Master doppio 2400K 9895A	14 002.000
Master singolo 1200K – opz 010	11 194 000
Trasformazione singolo/doppio 0868 2A	5.224.000
Dischi rigidi tipo Winchester	
Disco rigido da 4.8M Byte	
con floppy da 270K (3") 9133V	7.847 000
Disco rigido da 14.5M Byte	
con floppy da 710K (3.5") 9133D	8.458 000
Disco rigido da 14.5M Byte	
con floppy da 270K (3") 9133XV	8.495 000
Disco rigido da 14.5M Byte 9134D	7 683 000
Disco rigido da 14.5M Byte 9134XV	7 873 000
Sottosistema a nastro da ¼" per backup HP 150 9144A	7 964 000
Plotter formato A4 a due penne 7470A	2 723.000
Plotter formato A3 a sei penne – 7475A	4 709 000
Tavoletta grafica- 9111A	5.555 000
Stampante a getto d'inchiostro-grafica 80 col/150 cps	1.233 000
Stampante ad impatto-grafica 80 col. 80 car/sec. 82905B	2.089 000
Stampante ad impatto-grafica	251.000
80 Col/160 cps 82906A	2.005.000
Stampante Laserjet	
8 pag/min RS232 2686AB	8 860 000
Stampante ad impatto-grafica	
136 col/200 cps 2932A	8.123.000
Stampante ad impatto-grafica	
136 col/200 cps 2934A	7 119 000
Caricatore di fogli singoli 29340S	1.893 000
Stampante a margherita	
RS 232 40/32 cps 2601A	8.587 000
Trascinatore di fogli 26010A	1.389 000
RS232C 20cps 2602A	4 081 000
Interfaccia HP IB opz. 046	373 000
Trascinatore di fogli 26020A	872 000
Stampante termica	
80 col /120 cps – 2071A	2.974 000
Stampante termica grafica	
80 col/120cps (solo Serie 80) 2671G	3.721.000
Stampante termica grafica	
80col /120cps 2673A	5.463 000
Monitor da 9" per HP-86 82912A	741 000
Monitor da 12" per HP 86 – 82913A	816 000
Accessori per serie 80	
Modulo di memoria 16KByte per HP-85A 82903A	565.900
Modulo di memoria 64KByte 82908A	1 105 000
Modulo di memoria 128KByte 82909A	1 482 000
Scheda CP/M per HP 86/87 82900A	1.243 000
Cassetto porta ROM 82936A	113 000
Accessori per serie 200	
Modulo di memoria 256KByte – 98256A	1.869 000
Modulo di memoria 1MByte 98257A	8.386 000
System Monitor per Assembler ROM 82826A	741.000
ROM Plotter per HP-87 00087-15002	477 000
ROM Input/Output per HP 87 00087 15003	741.000
ROM per matrici per HP-87 00087 15004	477 000
ROM Assembler per HP-87 – 00087-15007	741 000
ROM MIKSAM (Indexed Sequential) 00087 18011	477 000
ROM Disco Elettronico per HP-86A/87 00087 15012	479 000
ROM Programmazione avanzata per 86/87 00087 15005	477.000
Interfaccia per serie 80	
HP IB 82937A	892.000
Seriale RS 232C – 82939A	892 000
GP/10 82940A	1 243 000
BCD – 82941A	1.243 000
Parallela tipo Centronics 82940A	741 000
HP IL 82938A	741 000
Data Link 82966A	1 482 000
Interfaccia per serie 200	
Controllore DMA a due canali – 98620B	1 121 000
Interfaccia HPIB – 98624A	761 000
Interfaccia Video a colori (cavi incl.) 98627A	2.265 000
Interfaccia Datacomm 98625A	1 124 000

HONEYWELL
Honeywell HISI Via Vida, 11 – 20127 Milano

Questar/M packaged system MC1/A 64Kb RAM 2 minifloppy 600+600 Kb + video tastiera cavo per stampante	5.950 000
Questar/M packaged system MC1/B (come MC1/A) + stampante Lina 32	7 456 000
Questar/M 9050 C 256 Kb + 2 minifloppy da 800 Kb + video grafico + Lina 32	11 750.000
Questar/M 9050D 256 Kb + minifloppy 600 Kb + disco 5 Mb + Lina 32 + video	18 150.000
Questar/M 9050B 256 Kb come 9050D ma il disco fisso da 10 Mb	17 650.000
Questar/M 9050F come 9050D ma disco fisso da 10 Mb (5 fissi + 5 mobili)	22.850.000
Unità disco addizionale 20 Mb (10 fissi + 10 mobili)	12 750.000
Memoria addizionale 256 Kb Ram	3.000 000
Unità calcolo aritmetico addizionale	2.100 000
4 linee V24 addizionali	1 600 000
Interfaccia IEEE 488	1 800 000

Multibus 786	1 600 000
2 floppy 8" 1024 Kb cass.	8 500 000
Posto di lavoro addizionale	2.990 000
microSystem 6/10 (configurazione di base) microprocessore micro 6 128 kb. 1 floppy disk 650 kb video tastiera	4 720 000
microSystem 6/10 256 KB RAM 2 floppy 650 KB video tastiera	6.216 000
microSystem 6/10 512 RAM 2 floppy 650 KB video tastiera	6.640 000
microSystem 6/10 256 KB RAM 1 floppy 650 KB 1 Hard disk 20 MB video-tast.	13 110 000
microSystem 6/10 512 KB RAM Hard disk 1 floppy disk 650 KB video tastiera	13 524 000
Modulo Addizionale Memoria 364 KB	1 600 000
P C D Opzione di emulazione MS DOS GW Basic CP/M 86	1.200 000
Stampante ASPI 10	1 300 000
Stampante ASPI 30	1 600 000

HONEYWELL HISI (Italia)
Honeywell Hisi
Via Tazzoli, 6 20154 Milano

STAMPANTI	
S11	900 000
L11	900 000
S31	1 300 000
L31	1 300 000
S11CQ	1.200 000
L11CQ	1.200 000
S31CQ	1.600 000
L31CQ	1.600 000
R32	1.700 000
L32	1.500 000
S32CQ	2.000 000
L32CQ	1.900 000
S38	3.700 000
L38	3.500 000
L111	1 050 000
L12CQI	1 400 000
L32CQI	1 900 000
34CQ	2.500 000

IBM
IBM Italia Via Pirelli, 18 Milano

Sistema 23 IBM 64 K byte 2.4 Mbyte su minidisco e stampante 60 sps	11.060 000
Configurazione. 64 K Byte 2.4 su minidisco + Stampante 160 sps 164 K	11.934 000
Sistema 23 versione ergonomica 64 K di memoria centrale fino a 2.4 Mbyte su minidisco stampante 80 sps	11.620 000
Sistema 23 versione ergonomica 64 K fino a 2.4 Mb su mini disco stamp. 160 cps	12 836.000

IBM
IBM Italia - Distribuzione Prodotti - Via Fara, 35 - Milano

Personal computer IBM	
64 K RAM 2 dischetti 360 K CPU + video tastiera stampante a 80 cps	5 942 000
Stesso modello precedente senza stampante e 1 minidisco da 360 k Mod XT	4 073 000
128 K RAM, 1 floppy 360 K, 1 disco fisso da 10 Mb video tastiera stampante 80 cps adattatore per comunicazioni asincrone Mod XT	8 480 000
128 K RAM, 2 dischetti da 360 K 2 dischi fissi da 10 Mbyte, stampante 80 cps, adattatore, video, tastiera	13.921.000
PC AT 256K 1 floppy 1.2 Mb + video monocromatico	8.383.000
PC AT 256K 2 floppy 1.2 Mb + video monocromatico	9.605.000
PC AT Mod. esteso 512K 1 floppy 20 Mb + video monocr.	11 765.000
PC AT 512K 1 floppy 1.2 Mb 40 Mb + video monocr.	14 762.000
DOS 3.0	145.000
Stampante	1 151.000
Video colori	1.628.000
Personal Computer Portatile 256K 2 floppy da 360Kb	5.584 000
DOS 2.1	145 000

ICL (GB)
ICL Italia S.p.A.
Centro direzionale Milanofiori - 20094 Milano

16 64K RAM 2 minifloppy da 800 K CP/M 8 Bit Basic 80	6.000 000
16-256 K TSM 2 minifloppy da 800K CCP/M Basic 16 Bit	5.000 000
16-512 K come 16 256	5 800 000
25 64K RAM 1 winchester 5 M + 1 minifloppy 800 K 8 Bit CP/M Basic 80	8.500 000
35 256K RAM winchester 10 M+1 minifloppy 800 K MP/M 8 Bit Basic 80	10.000 000
36-256K RAM-1 Winchester 10M+1 minifloppy 800K 16 Bit	8 700 000
36 512K come 36 256	9.580 000
Video + tastiera	1 100 000
8464 VIDEOCOLORI	3 700 000
Stampante 3181	1.080 000
Stampante 3184	1 550 000
Stampante 3185	2 400 000
Stampante a margherita	2 700 000

ICS Satron
ICS Satron Via della Balduina, 88 - 00136 Roma

Home computer Masters 5	399 000
Unità microfloppy MF 5	790 000
Basic G	130 000
Basic F	150 000
Esp. RAM 32 Kb	150 000
M23 mark 41 (128 K, video, 1 minifloppy da 1Mb)	4 900 000
M23 mark V color	8 730 000
M23 mark X (winchester 7.9 Mb) + 1 minifloppy 1Mb	10 400 000
M243 (182 K, video, 2 minifloppy da 720 K, multiutente)	9.900 000
M243 con disco 10 M	16 450 000
M243 con disco 20 M	20 450 000
Plotter 8 colori PL 200	2.404 000
Disco aggiuntivo 10 M a M243	8.000 000
Disco aggiuntivo 20 M a 243	10 000 000
Terminale intelligente RT 20 per M 243	3 400 000
M 343 (264 k, video color grafica 2 floppy×1,2 Mb)	15 000 000
M 88 MARK 41	9 300 000
M 60 MARK 41/color	8 900 000

I M S International
Sigesco Italia S.p.A.
Via Giulia di Barolo 22b 10124 Torino

5000 IS	
5IS 6 600 Kb minifloppy 5" + 6 Mb Winchester 5"	14 800 000
5IS 12 600 Kb minifloppy 5" + 12 Mb Winchester 5"	16 080 000
5IS 24 600 Kb minifloppy 5" + 24 Mb Winchester 5"	17 840 000
5000 SX	
5SX.12 600 Kb minifloppy 5" + 12 Mb Winchester 5"	13 580 000
5SX.24 600 Kb minifloppy 5" + 24 Mb Winchester 5"	15 400 000
5SX.32 600 Kb minifloppy 5" + 32 Mb Winchester 5"	19 050 000
8000 SX	
8SX.12 1 Mb floppy 8" + 12 Mb Winchester 5"	15.840 000
8SX.24 1 Mb floppy 8" + 24 Mb Winchester 5"	17 670 000
8SX.34 1 Mb floppy 8" + 34 Mb Winchester 8"	21.280 000
8SX.72 1 Mb floppy 8" + 72 Mb Winchester 8"	30 000 000

Nota: prezzo del dollaro 1900 lire

ITT
Condor Informatics Via Graziani, 8 20145 Milano

Sistema floppy 3030	da 4.600 000
Sistema a dischi 3030	da 8 500 000
Sistema multiutente	da 12 000 000

ITT FACE TELEINFORMATICA (Italia)
Face Teleinformatica Spa
Piazza Mercato 11 00040 Latina

3710 Z80 4 MHz, 64 K, 2 minifloppy da 320 K, CP/M	6 660 000
3713 Z80 4 MHz, 64 K, 1 winchester 6" 10 MB + 1 minifloppy 320 K, CP/M	9.700 000

JOYTECH (Taiwan)
Electronic Devices s.r.l.
Via Ubaldo Comandini, 49 00173 Roma

C Plus II A-48KByte, tastierino numerico alimentatore 5A, compatibile Apple	700 000
C PLus II B-64 KByte	730 000
CPlus II C64 KByte Z80 (Dual Processor)	500 000
C Plus II D-64 KByte, Z80 40/80 Colonne	800.000
Elite III - 64 KByte Dual Processor tastierino numerico tasti funzione Basic e CPM	890 000
STARTER 1 C Plus II A + Drive Controller + Driver Mitac meccanica Shugart 5" + Monitor Philips 12" TP 200 + Joystick autocentrante	1.230 000
STARTER 2 C Plus II C + Drive Controller + Driver Mitac 5" + Monitor Philips 12" TP 200	1 400 000
STARTER 3 SISTEMA UFFICIO C Plus II B o Elite III + 2 Drivers Mitac 5" + Drive Controller + Monitor Philips 12" TP 200 + Interfaccia grafica per stampante + Stampante grafica e letter quality 90 Colonne 120/cps + Corso Word Processing	2.750 000
Interfaccia per driver	79 000
Interfaccia grafica per Epson	127 000
Interfaccia parallela Centronics	79 000
RS 232	118 000
Vio card	83 000
16K Ram	68 000
Z 80	87 000
Language Card	90 000
80 Colonne Videx	127.000
80 Colonne con Switch	185 800
Pal card	105 000
Forth card	87 000
Integer card	58 000
6809 con Software e Manuale	300 000
Wild Card	110 000
Driver Mitac 5" meccanica Shugart compatibile Apple	460 000
Hard Disk Mitac 5/10 M Byte 5 M	3.200 000
con Adaptor per Apple e IBM 10 M	3 600 000
Driver Slim trazione diretta compatibile Apple	550 000
Tastiera Mak II 81 Tasti Multitech	250.000
Stampante Panasonic F/T grafica e letter quality 80 Colonne 120/cps. interfaccia parallela Apple	1 050 000
Stampante Panasonic grafica 80 Colonne 120/cps. interfaccia parallela Apple	950 000
Monitor Philips TP 200 12" fosfori verdi	160 000
Monitor Hantarex CTM 2000 12" fosfori verdi, ambra	200 000
Monitor Hantarex CTM 2000 15" fosfori verdi, ambra	305 000
Joystick a rosnapelo a 2 pulsanti per Commodore Vic 20 64K, Atari, ecc.	25 000
Joystick autocentrante per Apple	30 000
Joystick autocentrante per Apple con regolazione a 2 pulsanti, scatola metallo	52 000
Registratore computer cassette per Commodore Vic 20 e 64	60 000

JUKI (Giappone)
Telcom Via Matteo Civitali 75 20148 Milano

Juki 6100 (interfaccia parallela CTX) 20 cps 110 col	1.360 000
Juki 6300 (int. parallela) 40 cps 132 col.	2.350 000
I/F seriale RS232	160 000
Inseri. aut. fogli	850 000
Trascinamoduli bidir	280 000
Espansione buffer 2Kb	75 000

Nota: prezzo dello yen 7 lire

KOALA TECHNOLOGIES CORP (U S A)
TELAV International
Via L. Da Vinci 43 20090 Trezzano S.N (MI)

Tavoletta grafica KOALA per APPLE	264 000
Tavoletta grafica KOALA per Commodore 64 Disk	211 000
Tavoletta grafica KOALA per Commodore 64 su cartuccia	232.000
Tavoletta grafica KOALA per PC IBM	316.000
Tavoletta grafica KOALA per ATARI Disk	211 000

KONTRON MIKROCOMPUTER Gmbh (Germania)
Eledra 3S Spa Viale Elvezia, 18 20154 Milano

PSI 80 D/M2 - 64 K RAM, 2 minifloppy da 308 K	937 700

WDS 20E winchester 20 M rack 19"	10.275.680
DMA per floppy	425.425
CP/M 2.2	418.000

LOGITEK
Transpart S.p.A.
Corso Sempione 75 - 20145 Milano

Logitec FT 5001	890 000
Logitec WP 550	1.200 000

LORENZON (Italia)
Lorenzon Elettronica s.n.c.
Via Venezia, 115 30030 Oriago di Mira (VE)

SHIBE 16K	750.000
Espansione a 32K	100.000
Driver minifloppy 80K 40 tracce una faccia	740 000
Driver minifloppy 327K 80 tracce doppia faccia	1 180 000
Ram Disk con programma ausiliari/grafica	220 000
Stampante 80 col	700 000
Stampante 132 col.	1 580 000
Monitor 9" fosfori verdi alta def	240 000
Monitor 12" fosfori verdi	250 000

MANNESMANN TALLY
Via Cadamosto, 3 - 20094 Corsico (MI)

MT80 80 col. 80 cps interfaccia parallela	700 000
MT 160L - 80 Col - 160 cps grafica - interfaccia seriale o parallela	1 785 000
MT 180L 132 Col. 180 cps grafica interfaccia seriale o parallela	2.000 000
Caricatore automatico di fogli per MT 180L	801 000
Caricatore automatico di fogli per MT280	801 000
Introduttore automatico frontale di fogli singolo per MT280	801 000
MT 280 132 Col. 200 cps. int. seriale o parallela	2.415 000
Alimentatore di fogli per MT 400	1 430 000
MT 420L - 132 Col. - 200 cps. grafica - interfaccia parallela o seriale	3.430 000
MT 440L 132 Col. 400 cps. grafica interfaccia parallela o seriale	3.805 000
MT 440 versione OCR-A/OCR-B Barcode	3.960 000
MT 440F 132 Col. 400 Cps. Stampa a 4 colori grafica NLQ	4.085 000
FT 10 Terminale video, 24 linee, 80 colonne, fosforo grigio/verde interfaccia seriale	2.580 000
MT 600 600 lpm Interfaccia parallela	19.340 000
Interfaccia seriale per MT 600	710.000

Nota: Prezzi legati alle valute correnti.

MEMOTECH (GB)
Modular Electronics Via Britannia n° 29 00183 Roma

Tastiera 48 MTX 500 48 K RAM 16 K dedicata video - 32 K memoria utente	592.000
Tastiera 80 MTX 512 80 K RAM 16 K dedicata video 64 K memoria utente	799 000
Stampante DMX 80 stampante 80 cps. 80 col.	736.500
Floppy disk FOX single floppy disk 500 K compreso scheda controller per 8 drive	1 185.500
RS 232 doppia interfaccia RS 232 + bus-control per floppy-disk	199.000
Word processing card su ROM 32 K	238 000
ME 1 espansione RAM 64 K	299 000
ME 2 espansione RAM 128 K	499.000
RING ROM per collegamento tra MTX	338.000
Pascal Linguaggio Pascal su ROM 16 K.	338.000
UP1 sistema CP/M80 colonne colore per single floppy-disk con manuali e Newword Wordprocessing	762.000
UP2 floppy singolo da 500 K. (QUME) + ventola e filtro per raddoppiare FDX single-floppy	847 000
FDX D FDX double floppy disk 1000 K. comprendente FDX S + UP1 + UP2	2.541 500
FDX SD come FDX D ma con un silicon disk da 256 K. al posto di un drive	2.965 000
SILIDISK Silicon disk da 256 K.	1 189.000
HDX 20 hard disk 20 MByte + drive da 1 MByte + MTX A 8 megahertz	10.932.000

MICRO DESIGN
MICRO design s.r.l. Via Rostan, 1 - 16155 Genova

CFD001 controller floppy 5", 8"	270.000
BUS 12 porte	61.000
MRE memoria Ram eprom 32 K (8 K standard)	195 000
CVP 001 interfaccia video 80 × 24 (kit)	290.000
SMF scheda multifunzione	130.000
SER 101 interfaccia seriale RS232	120.000
OCS orologio calendario	135 000
IMU int. musicale	30 000
CFD 011 aggiunta doppia densità	330 000
CP/M	220 000
Basic 16 K	80 000
Drive per floppy 8" singola faccia	330 000
Drive per floppy 5" doppia faccia 80 tracce (mini slim)	820 000
Winchester 11Mb formattati con alimentatore HOST adapter controller software	3 400 000
Alimentatore SWITCHING (5+12 12V)	200.000
Alimentatore SWITCHING (24±12+5V)	394.000
Monitor 12"	220.000
CP/M user group (54 volumi) su disco 8"	25.000

MITSUI (Giappone)
TELCOM
Via Matteo Civitali 75 20148 Milano

MC 2100 80 col 120 c/sec. (I/F parallela-seriale)	1 105 000
MC 2200P 80 col 180 c/sec (I/F parallela)	1.200 000
MC 2200S 80 col 180 c/sec. (I/F seriale)	1.530 000
MC 4200P - 132 col - 180 c/sec. (I/F parallela)	1 700 000
MC 4200S - 132 col - 180 c/sec. (I/F seriale)	1.800 000

Nota: prezzo dollo yen 7 lire

MOTOROLA (U.S.A.)
Motorola S.p.A.
Milano Fiori Stabile C2 - Assago 20090 Milano

EXORset 163	12.951.000
EXORset 100	8.835 000
EXORset 100	8 487 000

MOUSE SYSTEM
Telcom srl Via M. Civitali, 75 20148 Milano

Controllore per cursore video terminali	
Mod. MOUSE × IBM PC	640.000
Mod. MOUSE - PAINT per IBM PC (soft. escluso)	740.000

Nota: prezzo per dollaro L. 2000

MULTITECH (Formosa)
Digitek s.r.l.
V. Valli, 26 - 42011 Bagnolo in Piano (RE)

MPF II Computer - tastiera + accessori	1.380 000
ZCC 3 scheda Z 80 × CP/M × MPF III	140.000
J S 3 joy stick/paddle × MPF III	56 000
MAK 2 sistema Acculeel compatib.	290.000
RS 3 interfaccia RS 232 x MPF III	140 000
FDI 3 Interfaccia x 2 disk D MPF III	140.000
F DDD doppio disk-drive x MPF III	1.250 000
10 x 3 sistema Kobac 10 Mb x MPF III	3.500 000
PC 511 MPF PC/511-128 K RAM- 1 FDD x 360	4.650 000
PC 512 MPF PC/512 128K RAM- 2 FDD x 360	5.200.000
PC 522 MPF PC/522 256K RAM 2 FDD x 360	5 550 000
PC 532 MPF PC/532 385 K RAM 2 FDD x 360	6.300 000
PC 542 MPF PC/542 640K RAM 2 FDD x 360	6.850.000
10 x PC sistema Xobec 10 MB x MPF PC.	3.500 000
PC x T51 MPF PC-xT/51 128K RAM	8.350.000
PC x T52 MPF PC-xT/52 256K RAM	9.700 000
PC x T53 MPF PC-xT/53 385K RAM	10.300 000
PC x T54 MPF PC-xT/54 800K RAM	11.000 000
CVM PC sistema col. basculante 13 x PC	1 400.000
MEB-PC1 scheda espans. da 128K x PC	520.000
MEB - PC2 scheda espans. da 256K x PC	1 030.000
MEB - PC 3 scheda espans. da 384K x PC	1 400.000
ACA-PC scheda con 2 RS 232 x PC	270.000
PBA-PC Personal Basic x MPF PC	110 000
MSD - PC MS DOS x MPF PC	80.000
MPF 2 Computer/accessori base + acc	550.000
TEM 1 sistema est. Multitech Mod. 1	80.000

TED 1 tastiera est. dattilo x MPF II	280.000
TED 2 tastiera dattilo x MPF II Mod. 2	220.000
RS 232 C interfaccia seriale x MPF II	160.000
ST 40 stamp. termica Multitech 40 C	480.000
12 MBV 12 F verdi A.R. A/riflesso	450.000
12 MBA 12 F ambra A.R. A/riflesso	450.000
12 DKV 1 12 F verdi A/riflesso CK	300.000
FDI 2 interfaccia x drive Disk N. 2	150.000
FDD 2 floppy disk drive slim line	450.000

NEC (Giappone)
Genius Computer s.r.l.
Via G. Corna Pellegrini, 24 25100 Brescia

8001 Z80 a 4MHZ 32KB RAM, 24KB ROM - tastiera alfanumerica	785.000
8011 32KB Connessione floppy RS232C	770.000
8012 32KB Connessione floppy 8SLOTS a disposizione utente	760.000
8031 2 minifloppy 5"1/4 Singola faccia, doppia densità 286 KB formatt.	860.000
8032 Unità aggiuntiva di 2 minifloppy 5"1/4	780.000
8031-2W Unità con 2 minifloppy 5"1/4 doppia faccia, doppia densità, 640KB formatt.	1.800.000
8041 Monitor 12" fosfori verdi	410.000
8043 Monitor 12" a colori	1.555.000
8012-02 Scheda espansione di 32KB	900.000
8045 Penna ottica Opzione per mod. 8000 e mod. 8800	885.000
8023 Stampante a matrice di punti 80 colonne 120 cps	1.300.000
8826 Stampante a matrice di punti 136 colonne 120 cps	1.980.000
8801 Z80 a 4MHz 64KB RAM da 32 a 88 KB ROM tastiera alfanumerica, interfaccia seriale RS232C Interfaccia floppy 8" e 5"1/4	2.200.000
8851 Monitor 14" a fosfori verdi ad alta risoluzione	590.000
8853 Monitor 14" a colori ad alta risoluzione	2.015.000
8831 Unità con 2 minifloppy 5"1/4 doppia faccia, doppia densità, 640KB formatt.	2.000.000
8832 Unità aggiuntiva di 2 minifloppy da 5"1/4	1.875.000
8881 Unità con 2 floppy 8" doppia faccia; doppia densità 2MB formattati	3.280.000
8882 Unità aggiuntiva di 2 floppy da 8"	3.150.000
8201 Personal Computer Portatile 8085 CMOS - 16 KB RAM 32 KB ROM Display a cristalli liquidi 40 × 8	1.560.000
8221 Stampante termica 20 colonne	340.000
8281 Registratore a cassette	260.000

OKI (Giappone)
Technitron Viale Milanofiori Pal. E/2 20094 Assago (MI)

Microline 82 A 80 col. 120 CPS	860.000
Microline 83 A 132 col. 120 CPS	1.460.000
Microline 88 132 col. 200 CPS Parallela	2.235.000
Microline 92 80 col. 160 CPS Parallela	1.185.000
Microline 84 132 Col. 200 CPS Seriale	2.435.000
Microline 92 80 Col. 160 CPS Seriale	1.375.000
Microline 93 132 Col. 160 CPS Parallela	1.770.000
Microline 93 132 col. 160 CPS Seriale	1.680.000
OKI 2350 132 col. 350 CPS	6.050.000
OKI 2410 132 col. 350 CPS 810	6.420.000

OLIVETTI (Italia)
Olivetti S.p.A. - Ivrea

M 10 24K RAM	1.740.000
M 20 ST versione bifloppy + esp. RAM 160K + stampante PR 1450	7.390.000
M24 bifloppy 256K RAM video	5.850.000
M21 come 20ST ma con 256K RAM Video monocromo	4.900.000
PL10 microplotter per M10	480.000
MC10 acoustic coupler per M10	800.000
Espansione 8K per M10	210.000

OLYMPIA (Germania)
Sogi Via Tenaro, 12 20124 Milano

Mod. ESW 102 KD Stamp. a margherita 17 cps incl. buffer 4K	2.200.000
Mod. ESW 103 KSK Stampante a margherita 17 cps incl. buffer 4K + tastiera	2.500.000
Mod. ESW 3000 BD Stampante a margherita 39 cps incl. buffer 4K	2.700.000

ONYX SYSTEMS INC (U.S.A.)
Strhold Sistemi EDP Via A. Cipriani, 2 42100 Reggio Emilia

C5011D/192KB/21MB/ 3 porte + Terminale Onyx	13.800.000
C5010V1/384KB/21MB (68MS)/11 porte	19.000.000
C5010V1/896KB/21MB (68MS)/11 porte	22.800.000
C5010V1/384KB/40MB (55MS)/11 porte	25.000.000
C5010V1/896KB/40MB (55MS)/11 porte	26.400.000
C5010VT1/896KB/40MB (30MS)/11 porte	32.900.000
C50120/512K/21MB/5 porte Terminale Onyx, Licenza UNIX System III	24.400.000
C50120/512K/40MB/5 porte Terminale Onyx, Licenza UNIX System III	29.500.000
C5012V1/512K/21MB/5 porte Terminale Onyx, Licenza UNIX System III	26.500.000
C5012V1/512K/40MB/5 porte Terminale Onyx, Licenza UNIX System III	31.000.000
C5012V1/1024K/21MB/11 porte Terminale Onyx, Licenza UNIX System III	31.000.000
C5012V1/1024K/40MB/11 porte, Terminale Onyx, Licenza UNIX System III	36.500.000
C5012V2/512K/2×40MB/5 porte Terminale Onyx, Licenza UNIX System III	37.500.000
C5012V2/1024K/2×40MB/11 porte, Terminale Onyx Licenza UNIX System III	43.000.000
Onyx Office (Word Processor Spreadsheet, Database Unify, Posta elettronica)	
Versione Runtime	3.900.000
Sistema di sviluppo	4.900.000

PHILIPS S.p.A.
Philips S.p.A.
Piazza IV Novembre, 3 20124 Milano

VG 8000 Computer MSX 32K RAM 32K ROM 32 sprites 16K video RAM	610.000
VG 8010 computer MSX 48K RAM	570.000
Monitor monocromatico BM 7552	180.000
Stampante a matrice 80 col. V W 0020	450.000
Stampante a matrice 40 col. V W 0010	260.000
Interfaccia stampante VV 0040 parallela Centronics	55.000
Espansione 16K RAM	72.000
G7400 Videopac con 2 joystick	100.000
C7420 Modulo estensione home computer	100.000
Registratore D6600/60P	90.000
Cartucce gioco	da 20.000 a 50.000

PHILIPS
LAGOSISTEMI S.p.A. - Via Torino, 2 20123 Milano

2093 1 Scheda 16 Bit con 128 Kb RAM con MS/DOS (dischetto e manuale)	800.000
2093 2 come 2093 1 ma con 256 Kb RAM	1.200.000
P2500/M CPU+Monitor grafico 12" 64 Kb RAM + 2 minifloppy da 304 Kb + tastiera italiana + interfaccia seriale CP/M	6.750.000
P2500/M DS come P 2500/M con 2 minifloppy da 604 Kb	8.150.000
P2010 Portatile con 2 minifloppy da 160 KB + CP/M + Wordstar e Calcstar	3.650.000
P2012 Portatile con 2 minifloppy da 640 KB + CP/M + Wordstar e Calcstar	4.600.000
P2145 disco fisso 10 Mb	3.450.000
P2092 Scheda di Esp. di memoria 256KB	850.000
P2091 Scheda IEEE 488 Comprensiva di IEEE 488 basic	800.000
PT 80 Stampante 80 Col 80 cps	1.000.000
PT180 Stampante 132 Col. 160CPS	2.000.000
P2900 083 Plotter a 3 colori	1.800.000

PHILIPS
SIGESCO Italia S.p.A. Via G. di Barolo, 22 bis - 10124 Torino

P 2010 Portatile con video 9" e 2 Floppy 5" da 160 Kb cadauno	
Con CP/M TESI (inf. retrieval) WordStar CalcStar	3.950.000
Con CP/M Scheda IEE 488 e Basic esteso	4.100.000
P 2012 Portatile con video 9" e 2 Floppy 5" da 640 Kb cadauno.	
Con CP/M. TESI (inf. retrieval). WordStar. CalcStar	4.890.000
Con CP/M, TESI WordStar MailMerge, CalcStar, InfoStar	4.950.000
Con CP/M Scheda IEE 488 e Basic esteso	4.980.000
Scheda 256 Kb aggiuntiva per Virtual Disk	850.000
Scheda 16 BIT 128 K RAM MS DOS	800.000
Scheda I/O controller IEE 488 e IEC Basic	800.000
Stampante 100 CPS 80 Col	950.000
Stampante 180 CPS 132 Col.	2.000.000
Disco rigido 10Mb	3.480.000

POLICONSULT Scientifica
Policonsult Scientifica s.r.c.
Via Pian due Torri, 65 00146 Roma

PCS 84000/8 2 drive 500 KB + video	6.600.000
PCS 84001/8: 2 drive 1 MB + video	8.100.000

PCS 84002/B 1 dr 500 KB + M.disk 5,8 MB + video	10.800.000
PCS 84003/B. 1 dr 1 MB + H disk 5,8 MB + video	11.100.000
PCS 84004/B: 1 dr 800 KB + H disk 10 MB + video	11.400.000
PCS 84005/B 1 dr 1 MB + H disk 10 MB + video	12.100.000
PCS 84006/B 1 dr 1 MB + H disk 20 MB + video	15.500.000
PCS 84007/B 1 dr 1 MB + H disk 28 MB + video	18.500.000
PCS 84000/C: 2 drive 500 KB + video	8.800.000
PCS 84001/C: 2 drive 1 MB + video	10.400.000
PCS 84002/C: 1 dr 800 KB + H.disk 5 8 MB + video	13.100.000
PCS 84003/C: 1 dr 1 MB + H.disk 5 8 MB + video	13.700.000
PCS 84004/C: 1 dr 500 KB + H.disk 10 MB + video	13.700.000
PCS 84005/C. 1 dr 1 MB + H.disk 10 MB + video	14.400.000
PCS 84006/C: 1 dr 1 MB + H.disk 20 MB + video	17.800.000
PCS 84007/C: 1 dr 1 MB + H.disk 25 MB + video	20.900.000
PCS 84002/BM 1 dr 500 KB + H.disk 5 8 MB + video	12.500.000
PCS 84003/BM 1 dr 1 MB + H.disk 5.8 MB + video	13.500.000
PCS 84004/BM 1 dr 500 KB + H.disk 10 MB + video	14.500.000
PCS 84005/BM. 1 dr 1 MB + H.disk 10 MB + video	16.800.000
PCS 84006/BM. 1 dr 1 MB + H disk 20 MB + video	17.500.000
PCS 84007/BM. 1 dr 1 MB + H.disk 27 MB + video	19.500.000
PCS 84008/BM Ogni posto di lavoro aggiuntivo	1.700.000
PCS 84002/CM 1 dr 500 KB + H disk 5 8 MB + video	14.800.000
PCS 84003/CM 1 dr 1 MB + H disk 5 8 MB + video	15.800.000
PCS 84004/CM 1 dr 500 KB + H disk 10 MB + video	16.800.000
PCS 84005/CM. 1 dr 1 MB + H disk 10 MB + video	17.800.000
PCS 84006/CM. 1 dr 1 MB + H disk 20 MB + video	18.800.000
PCS 84007/CM. 1 dr 1 MB + H disk 28 MB + video	21.800.000
PCS "Formula 1" CPUZ80 + 60KB + 2 Drive da 500 KB cad + Monitor 5" o 9" 80×64 fosfori verdi + Stamp. Termica 80 car + Programmatore di EPROM	5.800.000

PRINTONIX (U.S.A.)

M P H - Via General Govone, 56 - 20155 Milano

MVP	9.576.000
P300	13.503.000
P600	18.847.000
P300 XQ	16.932.000
P600 XQ	16.932.000
P600 XQ	22.431.000
Sottosistemi di stampa grafica per CPU IBM	

D.M.S.

M.P.H. Via General Govone, 56 - 20155 Milano

Magnum M 3000 Controller intelligente gestione grafica stampante Printonix	6.000.200
Magnum M 3400	8.757.100
Magnum M 2700	8.757.100
Magnum M 3276	8.757.100
Nota: 1$ = 1.900	

QUARK (Italia)

Quark s.r.l. Via Rasa, 32 20125 Milano

2010 - Z80/A - 64 K RAM - 1 disco 10 MB + 1 drive 1 MB	7.500.000
2027 Come 2010 ma con 1 disco da 27,5 MB	9.500.000
VISUAL 50 - terminale video 12" - 25 righe 80 colonne	1.800.000
3900 Posto di lavoro con tastiera. Video 15" CPU 64 KB RAM	2.800.000
3901 Come 3900 con tastiera 7 tasti funzione senza pad numerico	2.400.000
3125 - Unità Floppy 1 drive da 250 KB	1.200.000
3225 - Unità Floppy 2 drive da 250 KB	1.600.000
3150 - Unità Floppy 1 drive da 800 KB	1.400.000
3204 - Unità Floppy 2 drive da 800 KB	2.000.000
3101 - Unità Floppy 1 drive da 1 MB	1.600.000
3201 Unità Floppy 2 drive da 1 MB	2.400.000
3610 Unità Winch. 10 MB per 6 posti di lavoro + 1 drive 1 MB	8.800.000
3627 Unità Winch. per 8 posti di lavoro 27.5 MB + 1 drive da 1 MB	10.500.000

ROLAND

TELAV International
Via L. Da Vinci 43 20090 Trezzano S.N. (MI)

Mod DXY 101 Plotter 1 penna	1.147.000
Mod DXY 800 Plotter 8 penne	1.638.000
Mod DXY 880 Plotter 8 penne compatibili con HP 7470	2.108.000

SAGA (Italia)

S.A.G.A. S.p.A.
Via Vincenzo Bellini 24 00198 Roma

BIP 40/1 64 K RAM 1 Minifloppy da 400K + video 12"	3.900.000
BIP 40/2 64 K RAM 2 Minifloppy da 400K + video 12"	4.500.000
BIP 80 64 K RAM 2 Minifloppy da 800K + video 12"	5.600.000
BIP W 80 60 K RAM 1 Minifloppy 400 K 1 Winchester 9M + video 12"	7.450.000
BIP W 80 64 K RAM 1 Minifloppy 800 K 1 Winchester 9M + video 12"	8.600.000
BIP W 80X 256 K RAM 1 Minifloppy 800K 1 Winchester 9M + video 12"	9.950.000
Sagafox mod 40 - 64 K RAM 2 minifloppy da 410 K + video 12"	6.650.000
Sagafox mod. 80 64 K RAM 2 minifloppy da 800 K + video 12"	7.500.000
Sagafox mod. 40W6 - 60 K RAM 1 minifloppy 410 K + 1 winchester 5.8 M	10.900.000
Sagafox mod 80W6 - 64 K RAM 1 minifloppy 410 K + 1 winchester 8 7 M	11.450.000
Sagafox mod. 80 W 9 60 K RAM, 1 minifloppy 800K + 1 winch. 8.7 M + video 12"	12.450.000
Modulo FEM per Sagafox	800.000
Dual mod. 80W9 - 192 K RAM, 1 minifloppy 820 K + 1 winch. 8 7 M 1 p lavoro	14.000.000
Dual mod 80W6 - 192 K RAM 1 minifloppy 820 K + 1 winch. 8 7 M 2 p lavoro	16.000.000
Multi-F master con 1 minifloppy 410 K + 1 winchester 8.7 M	12.450.000
Posto senza minifloppy per Multi-F	3.000.000
Posto con 1 minifloppy per Multi-F	5.500.000
Posto con 2 minifloppy per Multi F	5.864.000
Sistema F9/6 unità con 2 microproc. Z80 A e Z80 B 256 K RAM. minifloppy 820 kb + 1 winchester 8.7 Mb 1 posto di lavoro con video tastiera separata	15.800.000
con due posti di lavoro	18.500.000
con tre posti di lavoro	21.100.000
Differenza disco fisso da 20 Mb	1.600.000

SCALA INDUSTRIAL CORP

Fabio Pagnani
Via Ubaldo Comandini, 49 00173 Roma

1048A Susy 2-48K/S	864.000
1064A Susy 2-80K/S	832.000
1000B Susy 2-64K/Z80	1.088.000
1000X Susy 2-64K/TC	1.224.000
1264 Susy 2-PC1	1.362.000
1364 Susy 2-PC2	1.139.000
1464 Susy 2-PC3	1.366.000
18016 Susy 5-PC/IBM	2.800.300
2048 Scheda Susy 2-48 zoccolata	381.000
2064 Scheda Susy 2-80 zoccolata	465.000
0001 Disk drive card	78.000
50010 dual side int. disk c.	121.000
6002 EPSON print int. card	73.500
6010 P print C	70.300
6052 Universal print int. c.	131.000
602 Printer cable	41.000
6003 Language card	102.300
6008 16K RAM card	100.000
6004 Integer card	102.800
6005 Z80 FP/M card	77.400
6000 80 colonne card	129.504
6006S 80 col. c. con switch	169.700
6007 RS232 card	109.100
6011 Communication card	109.160
6012 7718 Asyncronous s.c.	220.200
6013 Forth card	89.400
6014 Grapple c. 8 cable	178.400
6015 Buffer card	298.000
615 Buffer cable 2PCS	81.000
6016 6522 contr. card	84.800
6018 IEE488 c.	254.000
6020 Speech c. 8 speaker sw	80.100
6022 128K RAM card	805.000
6023 6809 card	358.500
6025 Music system c. 8 sp sw	120.000
6027 Wild c. 8 sw	84.700
6032 Pal card w/modulator	110.000
6035 AD/DA card W/SW	337.000
6039 Eprom writer 2716/32/64	127.300

6039 Clock e B SW	124 000
6040 Appletone card	357 000
6043 Olivetti int.	234 000
6048 IBM card (8088)	620 000
6051 RGB card B cable	131 000
6055 Appli Z80 + 64KRAM	648.000
6057 IC test card	255 900
4004 RF Modul & TV switch	17 800
4005 RF " con voice	18 800
4008 Joystick	23 500
4008T Desk top joystick	33 000
4009 Joystick autocentering	47 000
4009D Joystick doppio quick fire	55 400
4007 "Mouse" 91 functions	125 100
4014 Fan	29 500
4014P Cooling Fan & Cable	75 000
4015 "Light Pen" in ros.	393 400
4018 Graphic Table	123 000
4021 Extenderal port	10 000
4001 Keyboard per FP 1048	144 500
4001A Keyboard per FP 1048A	170 800
4001FK Keyboard per FK, 1064A,R	182 000
4002 Switching power S. 5V/3A	89 700
4002T Switching power S 5V/5A	85 500
4003 Case per FP1048	42 500
4003A Case per FP1488A, FK, FP 1064A,B	142.600
4203 Case e Keyb per 1364	510 000
4403 Case e Keyb per 1264 1464 IBM Keyb	583 000
4489 Tastiera PC IBM per APPLE	222 000
4491 Tastiera PC IBM per IBM	244.000
5000 5" Slim meccanica Multitech	421.000
5001 5" Slim meccanica Teac S/S SU	532.800
5002 5" Slim Scala D/S Susy 2	727 000
5003 8" Slim doppio drive Teac	1 042 000
5004 5" Slim meccanica Teac D/S Susy 2	532 800
6007 5" Slim drive per Apple	395 000

SEIKO
FOTOREX S.p.A
Via Fiume 48 20099 Sesto S. Giovanni (MI)

Serie 8600 per monoutenza composto da 128 KB 1 posto di lavoro 1 floppy da 655 Kb 1 Hard disk 10 Mb 1 esp. di memoria 128 Kb	14.950.000
Serie 4900 per multiutenza 128 Kb 2 posti lavoro 1 floppy 655 Kb 1 Hard disk 2 Esp. di memoria 128 Kb	18 800 000
Posto di lavoro supplementare	2.823.000

SEIKOSHA (Giappone)
Rebit Computer Divisione della GBC Italiana Spa
Viale Matteotti, 66 20092 Cinisello Balsamo (MI)

GP50A (40 Col. 40 CPS) int. parallela Centronics	260.000
GP50S (32 Col. 35 CPS) per Sinclair ZX81 e Spectrum	280 000
GP50 MX (40 Col. 40 CPS) per Computer MSX	280 000
GP55 AS(40 Col. 40 CPS) Interfaccia seriale RS 232C	330 000
GP100 AT (80 Col. 50 CPS) per Home Computer Atari	550 000
GP500 VC(80 Col. 60 CPS) per computer Commodore VIC 20 e 64	550 000
GP500 AS (80 Col. 50 CPS) int. seriale RS 232C	550.000
GP500 A (80 Col. 50 CPS) int. parallela Centronics	510 000
GP550 A (80 Col. 50 CPS) int. parallela Centronics & L.Q.	650 000
GP700 A(80 Col. 50 CPS) a colori int. parallela Centronics	850 000
GP700 VC (80 Col 50 CPS) a colori per Computer Commodore 64	800 000
SP800 A (80 Col. 80 CPS) & L.Q. int. parallela Centronics	740 000
SP800 I (80 Col. 40 CPS) & L.Q. versione totalmente PC IBM compatibile	790 000
BP 5200I (136 Col. 200 CPS) & L.Q. versione totalmente PC IBM compatibile	2.300 000
BP 5420A (136 Col. 420 CPS) & L.Q. int. parallela Centronics e seriale RS 232C	3 100 000
BP 6420I (136 Col. 420 CPS) N.L.Q. versione totalmente PC IBM compatibile	3.100 000
Interfaccia Grafica Apple II/GP500A	180 000
Interfaccia Grafica Apple II e Apple IIE/GP700A	219 000
Interfaccia Grafica Commodore 64/GP700A	190 000
Interfaccia Grafica Spectrum/GP700A	151 000
Interfaccia RS232C, TTY20 mA, TTL per GP700A	240.000

SHARP CORPORATION (Giappone)
Melchioni Computertime
Viale Europa, 49 Cologno Monzese - 20093 Milano

PA1000 - Macchina per scrivere portatile	839 000
MZ 731 Cpu MZ80A, 64K, cassetta 1200 Bit/secondi, stampante plotter 4 colori	1.260 000
MZ 3541/1 Cpu 2XZ80A, 128K RAM, 2 floppy disk, interfaccia parallela ed RS232C, video 12"	5.235 040
PC 5000/1 - Cpu 8086 16 Bit 128K RAM L.C.D 80x8 caratteri, stampante integrata bubble memory 128K	4 950.000
MZ 821 Home Computer CPU Z80 A 64 K Tastiera all. nuova cassetta magnetica 1200 Bit cps	930.000
MZ 5631/1 Personal Computer CPU 1 floppy disk M25631 + video fosfori verdi MZ1 013 + Tastiera (ISO 10M) MZ 1K 11 + Eos 16	5 858 000
10-0700 - Ink jet colour image printer	2.990 000

SIEMENS AG (Repubblica Federale Tedesca)
Siemens Elettra Spa
Via Lazzaroni 3 - 20124 Milano

Stampante PT88B aghi (80 cps. matrice 9×9)	1.200 000
Stampante PT48T ink jet (150 cps. matrice 9×9)	1 550 000
Stampante PT 898 ad aghi (80 CPS 132 col.)	1 850 000
Stampante PT 891 ink jet (150 CPS 132 col.)	1.800 000
Stampante PT80i2 ink jet (270 cps. matrice 8×12)	3.030 000
Stampante 2503 aghi (260 cps. matrice 8×7)	3.200 000
Stampante 2503 aghi (150 cps. matrice 8×8 OCR)	3.500.000
Stampante PT88 i ink jet (270 cps. matrice 9×12)	4.500 000
Lettore di caratteri OCR 2481 (apparecch. completo)	2.800 000

SINCLAIR (Gran Bretagna)
Rebit Computer G.B.C. Italiana S.p.A.
Viale Matteotti, 66 20092 Cinisello Balsamo (MI)

ZX Spectrum 16 K	339 000
ZX Spectrum 48 K	423.000
ZX Spectrum + 48 K - fornito in 18 versioni di lingue	550 000
ZX Microdrive	189 000
ZX Espansione System 80 K	340 000
Interfaccia 1	168 000
Interfaccia 2	80 000
ZX 81 con alim. 07 A	99 000
Espansione 16 K RAM Memotech	88 000
Kit di trasformazione.	
Per Spectrum 16K	148 000
Per Spectrum 48K	99 000
Interfaccia per monitor	60 000
Accessori e periferiche non di fabbricazione Sinclair	
Espansione 32 K RAM per ZX Spectrum 16 K con cassetta software	99 000
Penna ottica per ZX Spectrum	89 000
Box sonoro amplificato	25 000

SIPREL
Via di Vittorio 82, Zona Ind.le Baraccola 60020 Candia (AN)

KID 6410 (64K RAM RGB TV, drive 5" monitor 12" e tastiera)	2.090 000
KID 6820 (come KID 6410 con due drive 5")	2 820 000
KID 68SX (come KID 6410 con Z80 Winchester 10MB e back up incorporati)	8.850 000
KID 04FX (come KID 6410 con drive 5" 640KB)	4 800 000
Interfaccia parallela tipo Grappler	120 000
Interfaccia seriale RS 232	280 000
Scheda Z 80	220 000
Scheda 80 colonne con softswitch	300 000
Floppy disk 5" (scatola da 10)	50 000
Monitor PMG 12" (fosfori verdi)	220 000
Monitor PML 12" (fosfori gialli)	240 000
Trattamento antiriflesso per monitor PM	15 000
Monitor CM 14" (colori risoluzione standard)	550.000
Stampante Gemini 10x (ad aghi, 80 col. 120 car/sec. grafica)	936.000
Stampante Gemini 15x (ad aghi, 132col. 120 car/sec grafica)	1.348 000
Stampante Delta 10 (ad aghi, 80 col. 160 car/Sec. grafica buffer 8K, ingresso seriale e parallelo)	1 448 000

Stampante Delta 15 (ad aghi, 132 col, 160 car/sec, grafica, buffer 8K, ingresso seriale e parallelo)	1.896.000
Stampante Radix 10 (ad aghi, 80 col, 200 car/sec, grafica, buffer 16K, ingresso seriale e parallelo)	2.164.000
Stampante Radix 15 (ad aghi, 132 col, 200 car/sec, grafica buffer 16K, ingresso seriale e parallelo)	2.556.000

SPECTRAGRAM (U.S.A.)
TELAV International
Via L. Da Vinci 43 20090 Trezzano S.N. (MI)

Scheda 808 256 colori per Apple II	292.000

SPECTRAVIDEO (U.S.A.)
COMTRAD S.r.l.
Piazza Dante, 19/20 57100 Livorno

SVI 318 MK II Computer (64 K ROM/RAM)	445.000
SVI 328 MK II Computer (32 K ROM/80 K RAM)	802.000
SVI 904 Cassette Drive	118.000
SVI 602 Mini Expander	88.500
SVI 205 Cavo per stampante	87.000
SVI 802 Centronics Interface	174.000
SVI 803 16 K RAM	87.000
SVI 805 RS 232 Interface	174.000
SVI 807 64 K RAM	245.500
SVI 105 Tavola Grafica	108.000
SVI 603 Adattatore per Coleco con 2 Joystick SVI 103	164.000
SVI 010 Compumate	138.000
SVI 101 Joystick	20.500
SVI 102 Joystick	23.000
SVI 103 Joystick	32.700
Monitor colori 14"	828.000
SVI 728 MSX Computer (32 KROM + 80 KRAM)	665.000
SVI 707 MSX Disk drive 320 K	817.000
SVI 727 MSX 80 Column Card	205.000
SVI 737 MSX Modem con RS 232 Interface	327.000
SVI 747 MSX 64 K RAM	207.500
SVI 757 MSX RS 232 Interface	188.500
SVI 101/MSX Joystick	17.500
SVI 606 MSX Adattatore per SVI 318/328	237.500
SVI 808 MSX Modem	154.000
SVI 201 MSX Alimentatore per SVI 728/707	50.500
SVI 207 MSX Cavo per stampante	50.500
SVI 208 MSX Cavo per cassette drive	38.000
SVI 209 MSX Cavo RS 232 per SVI 737/757	77.000

STAR EUROPE
CLAITRON S.p.A.
V.le Certosa, 269 20151 Milano

STAMPANTI	
DP-8280 80 col 50 cps	490.000
DP 515 136 Col 100 cps	1.200.000
Gemini 10×80 col 120 cps	000.000
Gemini 15×136 col 120 cps	1.345.000
Delta 10×90 col 160 cps (200)	1.380.000
Delta 15×136 col 160 cps (200)	1.860.000
Radix 10×80 col 200 cps (240) NLQ	1.800.000
Radix 15×136 col 200 cps (280) NLQ	2.173.000
Powertype (Daisy Weel) 18 cps	1.114.000
STX 80 80 col 60 cps	800.000

TANDY RADIO SHACK (U.S.A.)
INFOPASS P.zza S.ta M. Beltrade, 8 20123 Milano
SECOR P.zza Primo Maggio, 36 33100 Udine
ITALSELDA Viale Cesare Pavese 45, 00144 Roma
SUPERTRONIC Viale Monza 226 20128 Milano
H.S.S. Via Carosso 11 98100 Messina

Mod. 100 8K	1.350.000
Mod. 100 24 K	1.860.000
Mod. 100 8K con Modem	1.528.000
Mod. 100 24K con Modem	1.870.000
Drive per Mod. 100 con interfaccia video	1.875.000
8K RAM per Mod. 100	180.000
Alimentatore per Mod. 100	21.000
Registratore CCR-81	114.000
Nino Color 16K Basic	380.000
Color 16K Extended Basic	440.000
Color 32K Extended Basic	650.000
16K RAM Color	111.000
32K RAM Color	244.000
Joy Stick	37.000
Drive 0 Nino Color	995.000
Drive 1 Nino Color	870.000
Mod. 4 16K	1.870.000
Mod. 4 64K 1 Drive	2.880.000
Mod. 4 128K 2 Drive con CP/M Plus	3.500.000
Mod. 4 64K 2 drive portatile	3.650.000
Mod. 4 High resolution	860.000
Hard Disk 5MB Mod. 3/4 Primary	5.424.000
Hard Disk 5MB Mod. 3/4 Second	4.407.000
Tandy 2000 128K RAM 2MB Comp. IBM	6.400.000
Tandy 2000 350Kb + H.D. 10MB	11.450.000
Monocromo per TANDY 2000	460.000
Color monitor per TANDY 2000	1.822.000
Exp. a 16 bit per MOD. 2	3.308.000
Mod. 12 80K 1 Drive	6.050.000
Mod. 12 80K 2 Drive	8.350.000
Drive 3 Slim Bay	2.087.000
Drive 3 e 4 Slim Bay	4.899.000
Mod. 168 256K H.D. 15 Mb con Xenix	15.990.000
Secondary H.D. 15 Mb	5.170.000
128K RAM CHIP Mod. 16	1.101.000
128K RAM Board Mod. 16	1.582.000
Scheda grafica	1.134.000
Drive 3 Slim Bay	2.087.000
Drive 3 e 4 Slim Bay	4.699.000
DT 1 Terminal	1.704.000
SNA/3270	1.400.000
BIS 3780	1.950.000
BIS-3270	2.000.000
CLEO 3270	1.950.000

TATUNG
Telcom srl Via Civitali, 75 20148 Milano

Terminale video mod. VT 4100	1.100.000
Terminale video mod. VT 4200 BASE	1.200.000
Terminale video mod. VT 4200 S Emulazioni Esprit II/ADM34/Viewpoint/DL80	1.300.000
Terminale video mod. VT 4200 B Emulazioni TVI 0126/TVI 920/TVI 925	1.280.000
Terminale video mod. VT 4200 C Emulazione ANSI 363/VT52/VT132 LIKE	1.260.000
Terminale video mod. VT 4200 D Emulazioni ADM 3A/Viewpoint/HZ1500/VT52	1.240.000

TAXAN
ELEDRA 3S s.p.a. Viale Elvezia n° 18 20154 Milano

KX1201 E (Monitor monocrom. 12" Fosfori verdi)	268.000
KX1212 E (Monitor monocrom. 12" per P.C. IBM)	390.000
KTS I (Piedistallo per monitor)	70.700
VISION EX (Monitor colori 12" RGB Video composito)	540.000
RGB VISION II (Monitor colori 12" RGB media risoluzione)	790.000
RGB VISION III (Monitor colori 12" RGB alta risoluzione)	860.000
RGB VISION PC (Monitor colori 12" RGB alta risoluzione per PC IBM)	860.000
CABLE I (cavo di connessione per PC IBM)	29.800
RGB IIE (Interfaccia monitor Apple II/IIE)	117.800
KP810 (Stampante 140cps 80 col. Bidirez. grafica)	1.153.000
KP910 (Stampante 140cps 156 col. Bidirez. grafica)	1.620.000
KPII (Interfaccia per stampante Apple II)	174.800

TELCOM
Telcom s.r.l.
Via Matteo Civitali 75 20148 Milano

Stampante TELCOM CPA 80P - 130 col - 130 c/sec. (I/F parallela)	765.000

Stampante TELCOM CPA 80S - 80 col - 130 c/sec (I/F seriale)	825.000
Stampante TELCOM CPB 80P - 80 col. - 130 CPS - int. parallela per IBM PC	800.000
Interfacce per adattamento stampanti YUKI MITSUI-CP80	
TC IFAPP01 Interfaccia adapter per Apple II IIe e compatibili	120.500
TC IFAPP12 Interfaccia grafica per Apple II IIe e compatibili	150.000
TC IFCC001 Buffer 8Kb esterno CTX - CTX	250.000
TC IFHP IB Interfaccia IEEE488 (HP IB) - CTX	180.000
TC IFSC001 Interfaccia seriale esterna RS232C 20mA - CTX (80b)	450.000
TC IFSC002 come sopra (2Kb)	300.000
TC IFC64 Interf. parall. CTX per Commodore, 64 e VIC 20	105.000
TC IF2CTX Sdoppiatore d'interfaccia CTX	195.000
TC IFSIBCL Interfaccia CTX per SPECT 16/48	120.000
Nota: prezzi per dollaro a L. 2.000	

TOBIA (Italia)
Ital. S.EL.DA.
V.le Cesare Pavese, 45 - 00144 Roma

10bC0 TOBIA BETA 1 B. 8" da 250 KB	3.480.000
10bC1 TOBIA BETA 1 B. 5" da 1MB	4.700.000
10bC2 TOBIA BETA 2 B. 8" da 2MB	6.000.000
109C1 TOBIA GAMMA 1 5" da 1MB cias.	6.000.000
109C2 TOBIA GAMMA 2 B. 5" da 1 MB tot.	8.800.000
10bC3 TOBIA GAMMA 2 B. 5" da 3,2 MB tot.	7.500.000
109C4 TOBIA GAMMA 2 B. 8" da 3,2 MB tot.	7.950.000
110C TOBIA DELTA con disco 5+5 MB	11.900.000
100C1 TOBIA PC Comp. IBM	3.500.000
100X1 TOBIA XT Comp. IBM	6.500.000
Controller DMTI 20C	816.000
Disco Winchester 65 MB	810.000
NDB 505 sottosistema DDB 5+5 MB per IBM PC	5.600.000
Unità disco 5+5 MB per micro	4.500.000

TOSHIBA (Giappone)
Tiber S.p.A.
Via Madonna del Riposo, 127 - 00165 Roma

81540 Tastiera con CPU 64K RAM C/INTERF	1.350.000
81542 Video verde 640×200 bits - grafico - 12" - 80 car ×25 linee	548.500
81643 Video colori 640×200 bits - grafico - 14" - 800 car ×25 linee	1.700.000
81548 Unità C/2 FDD - 560Kb×2	2.250.000
81647 Stampante grafica 80 col. 125 cps	1.550.000
81550 Stampante grafica - 132 col. 125 cps	1.880.000
81545 Cavo stampante	95.000
81548 Cavo Interf. RS 232/C	95.000
81548 Rom Pack DA BASIC	445.000
81553 Rom Pack - 16 Kb	445.000
81541 Cavo Alimentazione A/C	10.000
82577 Nastro stampante	25.000
82404 Sist. Op. CP/M	250.000
82405 Sist. Op. - DA/BASIC - DISK	120.000
82406 Sist. Op. T/BASIC - DISK	120.000
81513 - Personal Computer - T300 System 1	3.850.000
81501 Tastiera + CPU 192 Kb + Unità 2 FDD - 640 Kb×2+interf	4.800.000
81511 - Hard Disk 10 MB idee line	3.850.000
81506 Video verde - 12" - 640×500 bits - 80 chrs×25 linee	450.000
81505 Video colori 1 14" 640×500 bits - 80 chrs×25 linee - 8 colori	1.440.000
81503 Memoria addizionale 64Kb	230.000
81518 - Adattatore per comunicazioni	405.000
81519 - Date/Time clock	160.000
81507 Adattatore grafico - 1 - 640×800 per monocrom. (scheda)	460.000
81b04 - Adattatore grafico - 2 ×8 colori in Add. Adatt. Graf. 1	860.000
81516 - Adattatore Paletta per Video Colori n° 2 - 256 Colori	870.000
81547 Stampante 80 col. 125 cps - grafica	1.150.000
81550 Stampante 132 col. 125 cps - grafica	1.880.000
82410 Sist. Op. CP/M - 86 C/CBASIC - 86	565.000
82414 Sist. Op. MS DOS - 20 C/T BASIC 10	120.000
81234 Personal TOSHIBA T1100 - portatile video cristalli liquidi - 256 Kb RAM - 1FDD 3,5" 720 KB	8.225.000
81201 - T1500 System 2 - Comp. IBM - Hard e Software 2 floppy × 360 K - 128 KB RAM	3.700.000
81202 - T1500 come 81201 ma con 1 Hard disk 10Mb	8.700.000
Tastiera a basso profilo per T 1500	400.000
Video Monocromatico 12" 640×200 p. Per T1500	465.000
Video colori 13" 640×200 per T1500	1.180.000
Video cristalli liquidi 10" - 640×200 per T 1500	1.900.000

TRIUMPH ADLER (Germania)
Triumph Adler Italia S.p.A. - Viale Monza, 261 - 20126 Milano

Alphatronic PC	1.050.000
1a unità floppy per PC	1.000.000
2a unità floppy per PC	835.000
Video per PC	500.000
Alphatronic P2U 548 RAM + 16 ROM (64K utente) unità video 1920 caratteri a 2 unità minifloppy doppia faccia (2×320 KB)	4.200.000
Interfaccia IEC Bus	396.000
Interfaccia Parallela UPI 8606	585.000
Orologio Real Time EZU 8607	175.000
Full-Graphic CRT4A	1.080.000
Full-Graphic Term 1	1.400.000
P30 come P3 + microprocessore 80/88 + 64Kb + 128 Kb RAM	7.000.500
P40 come P4 + come P30	8.300.000
Alphatronic P3 come P2 U + 2 minifloppy da 1 Mb	6.460.000
Alphatronic P4 come P3 + 1 disco winchester da 5 Mb	8.500.000
DRH 80 stampante ad aghi 64 cps	1.450.000
DRH 130 stampante ad aghi 120 cps	1.930.000
DRS 250 stampante ad aghi 250 cps	3.300.000
TRD 170 stampante a margherita 17 cps	2.000.000
GARI 8008	605.000
Interfaccia parallela per Alphatronic PC e GARI 8008	200.000

V.D.S. (Italia)
DeDo sistemi - Piazza Indipendenza, 13 - 50100 Firenze

Eco 1 Z80A 64 Kb RAM video 28×80, tastiera separata, 1 floppy 8" doppia faccia 2,4 Mb, interfaccia seriale e parallela	9.500.000
Eco 2/7 come Eco 1 + Winchester 7 Mb e 1 floppy 8" 1,2 Mb	12.200.000
Eco 2/10 come Eco 2/7 con Winchester 10 Mb	12.800.000
Eco 2/16 come Eco 2/7 con Winchester 16 Mb	13.800.000
Eco 2/27 come Eco 2/7 con Winchester 27 Mb	14.800.000

VECTOR GRAPHIC (U.S.A.)
ORGOS srl - Via Andrea Ferrara, 12 - 00165 Roma

Sistemi 8/16 bit - Z80/8088.	
VECTOR 4/00 IWS - Intelligent Workstation per LINC	5.550.000
VECTOR 4/10 1 Minifloppy 630 Kb	6.850.000
VECTOR 4/20 2 Minifloppy (1,2 Mb)	7.750.000
VECTOR 4/30 Hard disk 5 Mb Minifloppy 630 Kb	10.650.000
VECTOR 4/40 Hard disk 10 Mb Minifloppy 630 Kb	12.650.000
VECTOR 4/80 Hard disk 36,6 Mb Minifloppy 630 Kb	21.700.000
VECTOR VSX 8/16 BIT Z80 8/8086 (8MHZ)	
VSX 2000 2 minifloppy (2×737Kb)	8.450.000
VSX 3000 H.D. 10Mb - minifloppy 737Kb	13.450.000
VSX 5000 H.D. 36,6Mb - minifloppy 737Kb	21.900.000
SISTEMI MULTIUTENTE:	
VECTOR 5005/E1 1 Hard disk 5 Mb 1 Minifloppy 630 Kb	12.950.500
VECTOR 5010/E1 1 Hard disk 10 Mb 1 Minifloppy 630 Kb	14.750.000
VECTOR 5032/E1 1 Hard disk 32 Mb, 1 Minifloppy 630 Kb	23.950.000
ACCESSORI	
Terminale Intelligente Aggiuntivo (Con Scheda 64 Kb RAM Scheda flashwater cavo di collegamento 5 m.)	3.450.000
TD 10 Sistema Tape (Per copie sicurezza su nastro)	6.900.000
Incremento da 128 a 256 Kb Ram per Vector 4/xx	1.280.000
Linc: Sistema per rete locale fino a 32 sistemi Vector 4/xx	1.340.000
Stampante Vector 3500 a margherita 35 CPS	4.500.050
Stampante Vector 7700 a margherita 55 CPS	6.200.000
Nota: prezzi per dollaro = L. 1.800	

VICTOR TECHNOLOGIES (U.S.A.)
HARDEN ITALIA S.p.A.
Milano Fiori - Strada n° 7 Palazzo T3 - 20089 Rozzano (MI)

H00001 Computer Victor 1 128K RAM	6.200.000
H00002 Computer Victor 1A 256K RAM	7.900.000

H00003 Computer Victor 18 256K RAM	10 800 000
H00005 Computer Victor (portatile) 256K RAM	6 500 000
H00026 Pacch V18 + MT 180 L + H Azienda	16 500 000
H00043 Pacch V18 + MT 180 V + S O Concurrent	14 000 000
H00004 Harden 10 MB - Completo - per V1/V1A	3 080 000
H00006 Harden 10 MB Senza D M A per V 18	3.080 000
H00035 Dminet transporter board L A N	1.250 000
H00030 Modulo di connessione - L A N	400 000
H00037 Network station - L A N	8 000 000
H00088 File server 256K 10 MB winch. - L A N	10 000 000
H00154 Harden 10 + 10 MB Victor Apricot	9.980 000
H00155 Harden 8 + 8 MB Victor - Apricot	8 980 000
H00018 Scheda Xebec controller	1.512 000
H00018 Scheda DMA card	1.075 000
H00021 Scheda espansione 128K - Victor	740 000
H00022 Scheda espansione 384K - Victor	1 080 000
H00025 Scheda co processore Arite 8087	860 000
H00044 Sistema CPM/80	1 380 000

VIDEO TECHNOLOGY (Hong Kong)
Compitent - Via Vittorio Emanuele 9 - 91021 Campobello di Mazzara (TP)

Mod Laser 200 pacchetto con 08K RAM 280 + 18 K microsoft basic (TRS 80 comp.)	490 000
Mod Laser 310 come Mod 200 più 1 drive 5"1/4 + interfaccia parallela	990.000
Mod Laser 2001 microprocessore 6502 a 2 MHZ + 48 K RAM grafica 280×192 + Microsoft esended basic + interfaccia parallela	560 000
Drive per Mod 2001 completo di controller	580 000

XEBEC INTERNATIONAL (U.S.A.)
Tre Pal s r l Via Asmara, 72 Roma

Mod. 9710 Sottosistema a dischi Winchester da 10 MBytes con adattatore Apple per DOS CPM Pascal; Prodos	3 400 000
IBM PC	3 400 000
Multibus per CMP-86	3 850 000
QBus per emulazione RL02/01	4.830 000
S 100	3 850.000
Mod 9730T Sottosistema a dischi Winchester da 30 MBytes con adattatore Apple per DOS CPM. Pascal Prodos	13 550 000
IBM PC	13 550 000
MULTIBUS per CPM 86	14 100 000
S 100 per CPM 80	14 100 000
QBus per emulazione RL02/01	14 900 000

XEROX CORPORATION (U S A)
Elu S.p.A. Via Imperia, 2 20142 Milano

XEROX 16/8, 2 floppy 8" da 1200 Kb (DS)	7 000 000
DIABLO 630 stampante a margherita 40 cps	4 750 000
DIABLO 620 stampante a margherita 22 cps	2.800 000
Sistema operativo CP/M	300 000
ACL X Chiave e compilatore ACL per compatibilità programmi DIABLO	680 000
XEROX 3050 64 Kb RAM video 24×80 2 floppy 8" da 1200 Kb (DS)	11 800 000
XEROX 3150 64 Kb RAM video 24×80 1 floppy 8" da 1200 Kb, hard disk 10 Mb	16 800 000
MUPA adattatore a 4 porte per collegamento posti di lavoro aggiuntivi	1.600 000
TV X Terminale video 24×80 con tastiera separata	3.000 000
XEROX XR Disco Rigido 10 Mb 1 floppy da 5" ¼ 340K	900 000
XEROX XF 64KB 2 drive da 5" ¼ 2×340K	5 700 000

YEW (Giappone)
Telcom - Via Matteo Civitali 75, 20148 Milano

PL 1000 mod. 0851 01 (interf parallela)	1 560 000
PL 1000 mod 0851 02 (interf seriale)	1 080 000
PL 1000 mod. 0851 11 (int parall. + ROM graf)	1 830 000
PL 1000 mod. 0851 12 (int. seriale + ROM graf)	1 930 000

Nota: prezzi per 1 yen 6.7 lire

SCHEDE A MICROPROCESSORE

A.S.EL. (Italia)

A.S.EL. s.r.l.
Via Cortina d'Ampezzo, 17 - 20139 Milano

Amico 2000 montato	335.000
Amico 2000 in kit	254.000
Alimentatore	16.500
Espansione BUS	127.000
Alimentatore di potenza montato	248.000
Contenitore con alimentatore di potenza, montato	603.000
Interfaccia video montata	234.000
Tastiera ASCII montata	264.000
Tastiera ASCII in kit	129.000
Scheda RAM/ROM Basic montata	299.000
Scheda RAM/ROM Basic in kit	780.000
Sistema completo Amico 280	1.680.000
A036 Scheda CPU	218.000
A033 RAM dinamica 16K	280.000
A033 RAM dinamica 32K	340.000
A034 Floppy disk controller	325.000
A027 Scheda Video	234.000
A035 Interfaccia SEG/PAG	174.000
A030 Scheda RAM/ROM	180.000
Mother Board	166.000

COSMIC (Italia)

Cosmic s.r.l. - Largo Luigi Antonelli, 2 - 00145 Roma

FDC/2 floppy disk controller	450.000

MOTOROLA (U.S.A.)

Motorola S.p.A. - Via Ciro Menotti, 11 - Milano

MEX M68 705 EVB	1.108.000
MEK 6802 D5 E	425.000

ROCKWELL INTERNATIONAL (U.S.A.)

Dott. Ing. Giuseppe De Mico S.p.A.
V.le Vittorio Veneto, 8 - Cassina de' Pecchi (Milano)

AIM 65/40 piastra base singola	1.822.000
AIM 65/40 versione completa periferiche (16 K)	3.228.000
AIM 65/40 versione completa periferiche (32 K)	3.778.000
Tastiera	234.000
Display	588.000
Stampante	770.000
AIM 65 1 K RAM	1.185.000
AIM 65 4 K RAM	1.256.000
Assembler 4 K	117.000
Basic 8 K	160.000
Forth 8 K	201.000
PL-65 8 K	204.000
Pascal 20 K Ram	335.000
Alimentatore 32	110.000
Espansione 32 k dinamica	726.000
Espansione 16 k PROM/ROM	281.000
Espansione 8 k	720.000
Programmatore di EPROM UNIVERSALE	280.000
Interfaccia video	730.000
Floppy disk controller	871.000
IEEE 488	723.000
Nota. Prezzo del dollaro e L. 1.650	

SGS ATES (Italia)

SGS ATES Componenti Elettronici S.p.A.
Via Carlo Olivetti, 2 - 20041 Agrate Brianza (Milano)

NBZ 80 - WW	650.000
NBZ 80-F	650.000
NBZ 80-G	650.000
NBZ 80-I	650.000
NBZ 80-S - WW	1.140.000
NBZ 80-S F	1.140.000
NBZ 80-S G	1.140.000
NBZ 80-S I	1.140.000
NBZ 80-S UK	1.140.000
NBZ 80-S US	1.140.000
NBZ 80-HL WW	2.070.000
NBZ 80-HL F	2.070.000
NBZ 80-HL G	2.070.000
NBZ 80-HL I	2.070.000
NBZ 80-HL UK	2.070.000
NBZ 80-HL US	2.070.000

TOBIA (Italia)

Ital S.EL.DA - V.le Cesare Pavese, 45 - 00144 Roma

501S Tobia CPU	195.000
502S Tobia Prom Basic COS	290.000
503S Tobia Ram Statiche 16K	320.000
504S Tobia Ram Dinamiche 48K	280.000
505S Tobia Configurator (anche su commessa)	330.000
506S Tobia CRT Video (con Prom da 48)	330.000
507S Tobia Controller Cass. Digitali	100.000
508S Tobia Bus	220.000
509S Tobia RS232	300.000
510S Kit assemblaggio Tobia Professional	600.000
511S Kit con 2 unità a cassetta digitali	290.000
512S Tobia Acoustic Coupler	300.000
513S Tobia Controller Cass. Audio Mod. 1	108.000
514S Tobia Controller Floppy Disc	380.050
515S Tobia Grafica	280.000
516S Tobia Caricatore Prom	220.000
517S Tobia Color 14	430.000
518S Tobia Monitor 12"	280.000

CALCOLATRICI PROGRAMMABILI E POCKET COMPUTER

CASIO (Giappone)

Ditron S.p.A. - Viale Certosa, 138 - 20156 Milano

PROGRAMMABILI	
FX 180 P	60.700
FX 3600 P	88.900
FX 4000 P	148.700
FX 190	61.400
POCKET COMPUTERS	
PB 110	118.000
PB 200	145.100
PB 410	180.500
FX750/P	280.800
PB700	383.700
ACCESSORI	
OR 1 (espansione per PB 110)	47.700
FA 3 (interfaccia PB 110/200/410)	80.600
FP 12 (stampante per PB 110/200/410)	153.500
FA 10 (interfaccia + plotter per PB700)	810.000
CM 1 (registratore per PB 700)	174.000
OR 4 (espansione per PB 700 4K)	95.000
FA 4 (interf. Centronics per PB700)	142.800
FA 20 (interf. Stamp. per PB750/P)	205.000
RC 2 (RAM CARD per PB410/FX750/P 2K)	68.500
RC 4 (RAM CARD per PB410/FX750/P4K)	108.000

HEWLETT PACKARD (U.S.A.)

Hewlett Packard Italiana - Via G. Di Vittorio, 9 - 20063 Cernusco sul Naviglio (MI)

Scientifica programmabile mem. perm. - HP-11C	178.000
Finanziaria programmabile mem. perm. HP 12C	288.000
Scientifica programmabile mem. perm. HP-15C	288.000
Programmabile per progettisti sist. HP-16C	288.000
Calcolatore alfanum. mem. perm. 310 reg. HP41CV	505.000

Calcolatore alfanum. mem. perm. 319 reg HP-41CX	748 000
Lettore di schede magn per HP-41 82104A	480 000
Stampante per HP-41 82143A	825.000
Lettore ottico per HP-41 82153A	300 000
Memoria di massa a cartuccia HP IL 82161A	1.315 000
Interfaccia HP-IL/RS232C 82164A	704 000
Interfaccia HP-IL/GP10 82165A	709 000
Kit Interfaccia HP IL 82166S	948 000
Interfaccia HP-IL/HP IB 82169A	837 000
Computer portatile HP 71 82	1.282 000
Computer portatile HP 75 DZ	2.817 000
Accessori per HP 71B	
Lettore di schede 82400A	394 000
Interfaccia HP IL 82401A	291 000
Modulo di memoria RAM (4K) 82420A	173.000
Accessori per HP 71B	
Lettore di schede 82400A	394 000
Interfaccia HP IL 82401A	291 000
Modulo di memoria RAM (4K) 82402A	173 000

SHARP (Giappone)
Melchioni S.p.A. Via P Colletta, 37 Milano

PC 1212 (programmabile in Basic)	198.000
CE 122 (stampante per PC-1211)	240.500
PC 1251	253 000
CE 125 (Unità con microcassette e stampante per PC-1251)	348 800
PC 1500/A	428 000
CE 150 stampante	450 000
CE 151 (espansione 4K per PC 1500)	135 000
CE 152	105 500
CE 155 (espansione 8K per PC 1500)	280 000
CE 158 (interfaccia seriale RS 232 e parallela per PC 1500)	398.000

TANDY RADIO SHACK
INFOPASS P.zza S.ta M Beltrade, 8 20123 Milano SECOR P.zza P Maggio, 36 33100 Udine
ITALSELDA Viale C. Pavese 45, 00144 Roma SUPERTRONIC Viale Monza 226 20128 Milano
H.S.S. Via Cernaia 11 98100 Messina H.S.S. Via Cernaia 11 - 98100 Messina

PC 2	410 000
Printer per PC2	437 000
4K RAM per PC2	125 000
8K RAM per PC2	242 000
Porta RS 232 C per PC2	436 000
PC4	188.000
Interfaccia reg per PC4	95.000
Printer per PC4	173 000
1K RAM PC4	35 000
Valigetta per PC2	65 000
PC3	235 000
Printer/cass per PC3	244.000

TEXAS INSTRUMENTS (U S A.)
Texas Instruments Semiconduttori Italia S P.A.
Divisione Prodotti Elettronici Personali Viale delle Scienze 02015 Cittaducale (Rieti)

TI 53	48 000
TI 55 II	78 000
TI-57 LCD	88 000
TI 66	98 000
PC-200	148 000
Biblioteche S.S.S (in italiano) ing. civile topografia	56.000
Biblioteche S.S.S (in inglese)	29 000
TI 30 GALAXY	39 000
TI 30 III* GALAXY	29 900

MC

Annunci gratuiti per vendita o scambio di materiale usato o comunque in unico esemplare, fra privati. Vedere istruzioni e modulo a pag. 193. Per motivi pratici, si prega di non lasciare comunicazioni o chiedere informazioni (telefoniche o scritte) riguardanti gli annunci inviati.

Vendo

Vendo causa passaggio a sistema superiore, **VIC 20 + Espansione 8 K RAM + 3 Cartridge** (Gorf, Alien, Omega Race) + 2 libri "Alla scoperta del VIC 20" e "Giochi per il vostro VIC 20". Valore complessivo L. 360 000, a L. 250 000 trattabili. Tratto soltanto con zona Torino. Telefonare 011/697463 chiedendo di "DINO"

Vendo **Macintosh + Image Writer** (entrambi ancora in garanzia) + numerosi programmi tutti originali e con manuale (Mac Draw Multiplan - Chart - Contabilità Generale - PFS File - e tanti altri) a lire 7 000 000 intrattabili. Telefonare allo 011/713766 o scrivere a Carrozzo Carmelo - Via Genova 7 - 10095 Grugliasco (TO)

Vendo Pocket Computer **TRS 80 PC-2** (identico allo **Sharp MC-1500**) completo di alimentatore e manuali (in inglese) a sole L. 400 000. Luca Favaro - V. Magnano 2 - 10010 Piverone (TO) - Tel. 0125/72279 ore 14 in poi

Vendo **Sharp MZ-731 (Registratore, stampante, plotter 4 colori incorporati)** Z80 A 64 K RAM + 4VRAM, 4KROM + 2K ROM (Gen. caratteri), tastiera ASCII, terminale RGB composito TV + Basic V1 CA + software vario + giochi L.M. prezzo corrente 1 250 000 + IVA (18%) il tutto a L. 900 000. Rivolgersi a David Alemagni · Via Torino n° 54 10013 Borgofranco · Tel. 0125/751040 Ore pasti e dalle 20 30 alle 23

Causa doppio regalo **vendo Pocket Computer Casio PB 700** nuovissimo mai usato con garanzia in bianco a L. 350 000 (nuovo costa 440 000 circa 20% in più). Disposto anche a vendere interfaccia FA-10 e registratore CM-1 in blocco con PB 700 a L. 1 000 000 (nuovo circa L. 1 300 000) causa scarso interesse. Tel. 0131/94265 ore pasti chiedere di Fabrizio

Vendo **Sharp MZ 700 + registratore + video 12" f.v.** + inediti per l'Italia (acquistati all'estero) floppy disk 3" interfaccia ROM 16 K + nuovo Basic più potente + vari manuali + 2 scatole di dischi + vari programmi. Tutto L. 1 400 000 oppure separatamente. Martini Marco - Via Don Perduca 51 - Tortona (AL) · Tel. 0131/867219

Vendo **Zx Spectrum 48K + Joystick con interfaccia programmabile + Lettore/registratore dati Philips d6600/30 p** + libri e manuali per la programmazione + moltissimo software ad un prezzo trattabile. Anche separatamente oppure scambio con altro computer (preferibilmente CBM 64 con unità di massa, sono disposto a conguaglio se floppy). Per informazioni 0173/95039 (dopo le 20)

Vendo in occasione **Vic 20 + Registratore 1530 Model C2N** + 7 Cartridge «Raid on fort Knox - Cosmic Cruncher + Home Baby Sitter - Drav Poker - Cartouche · Visible Solar Sistem - Gorf» + Joystick + 2 libri «50 esercizi in Basic - Il libro del Commodore Vic 20» + 50 giochi su cassette + riviste tutto 3 mesi di vita. Vendo solo in blocco L. 500 000 non trattabili. Belletti Walter - Via Marconi 11 - 28068 Romentino (NO)

Commodore Vic 20 + Registratore Commodore + Joystick + Espansione Grafica a di Memoria + 2 Cartridge + 80 programmi su cassette comprendenti quanto di meglio pubblicato e venduto + manuale in italiano + 2 libri sul Basic + riviste. Vendo causa passaggio a sistema superiore, tutto nuovo vera occasione L. 300 000 - Novara - Tel. 0321/451891 (Giuseppe)

Vendo **Zx Spectrum 48K** con poco più di 1 anno, usato pochissimo, con cavetti e alimentatore e manuale in italiano + cassette con programmi + 3 microdischi con programmi + cassette vergini + cassetta horizons + microdisco dimostrativo + interfacce 1 e 2 + microdrive con manuali + joystick quickshot 2 + 1 libro di programmi + alcune riviste di programmi per i sinclair, tutto a lire 850 000. Bavagnoli Ariberto - Corso S. Martino 8 - 13100 Vercelli · Tel. 0161/62012

Vendo **Apple II e 2 disk driver + stampante + monitor + scheda 80 col. e 64 K aggiuntivi** + scheda Z 80 + 400 programmi + altri accessori. Prezzo da concordare. Marco Verdesca - Res. Archi Milano 2 Segrate (MI) · Tel. 02/2138347

Vendo **Apple II Europlus con minuscole, language Card, Monitor 12", Controller e primo Disk Drive originali, Disk Drive aggiuntivo** scheda Z80 per CP/M, scheda parallela Centronics e 100 dischetti di marca con software di valore a lire 2.700 000 trattabili. Vendo anche solo parte configurazione. Il tutto è in perfette condizioni. Telefonare a Lamberto - Tel. 02/2849928

Vendo **ZX Spectrum 48 K completa di accessori, registratore** istruzioni in inglese e italiano varie cassette utility e giochi, tutto praticamente inutilizzato per L. 400 000. Giorgio Apostolo - Via Ampère 49 - 20131 Milano - Tel. 02/292732

Modulo 82180A Interfaccia HPIL per HP41 L. 200 000 Trattab. **Modulo Timer 82/82A per HP41** L. 100 000 - Sergio Bruschi - Tel. 02/321944 - Via Albani 52 - 20148 Milano

Vendo **Monitor Philips PCT 1201** 12 pollici, fosfori gialli usato solo un anno completo di schermo antiriflesso a L. 200 000 trattabili. Francesco Pezzino Via Console Marcello, 25 - 20156 Milano - Tel. 02/365365

Occasionissima **ZX Spectrum 48 K + Registratore + Interfaccia Kempston** + interfaccia programmabile Ten Kołek + oltre 800 programmi con le ultime novità inglesi + Joystick + numerose recenti riviste di Computer. Vendo causa passagio Q L. a lire 700 000 poco trattabili. Telefonare ore pasti a Riccardo - Tel. 4039788 (sola zona di Milano)

Vendo ZX 81 + espansione 16K + manuale per l'uso + libri con programmi a L. 150 000 inoltre vendo **Base Atari 2600** con 13 bellissime cassette a L. 200 000 oppure permuto il tutto con ZX Spectrum 48 K. Telefonare Andrea 02/4073143

Vendo Texas TI 99/4A completo di cavo registratore, modulatore, Joystick, registratore - Modulo SSS Schacchi + 12 cassette e riviste varie a L. 280 000 Aniello Giordano · Via del Pettirosso 10 · Milano Telefonare dalle 20 00 alle 21 00 allo 02/4151931

Vendo **Sharp MZ 700 con registratore e alimentatore incorporati** + valigetta per trasporto + cavo TV e alimentazione + 2 tipi di Basic + Programmi vari + ottimo manuale perfettamente idoneo per chiarire la programmazione Basic e L.M. + piccolo T.V. - causa cambio sistema vendo a L. 500 000 trattabili Bruno 02/417537

Occasione!! **Vendo TI-99/4A completo di accessori** + modulo SSS Extended Basic + 2 Joysticks + manuale e libro "L'Home Computer TI-99" + Modulo SSS Invaders e raccolta di oltre 80 programmi vari su nastro tutto a L. 290 000 · Telefonare ore 20-21 ad Andrea 02/426104 - Milano

Vendo ZX81 + Espansione 64K + Espansione grafica + Assembler in ROM + istruzioni in inglese e italiano + libro "66 programmi" + 2 libri in inglese sul BASIC e L/M + Cavetti + Circa 200 Programmi su cassetta e listati a sole L. 500 000. L'affare è assicurato in quanto il costo di tutto supera di molto il milione di lire. Scrivere o telefonare dopo le 20 a Francesco Adornato · Via C. Dentato 11 - 20146 Milano · Tel. 02/470372

Vendo Intellivision usato pochissimo con cavetti e imballo originali + 5 cartucce (tron, triple action, soccer, venture, mouse trap) in ottimo stato. Il tutto a L. 300 000 (telefonare dopo le 14). Righele Alberto via Livenza 6 36015 Schio (VI) tel. 0445/ 28879

CBM 4032 + 4040 + 3022 + amplificatore sonoro + lavagna pneumatica per listati con 3000 programmi - gestionali - ing. word proc. - visicalc su eprom - compilatori e tantissimi altri - vari manuali d'uso e tecnici. Tutto il materiale è praticamente nuovo - prezzo d'occasione. Sonagere Sergio - via Kennedy 35 · 33038 San Daniele del Fr. (UD) · Tel. 0432/955666 ore ufficio

Occasionissima - vendo **commodore/64 drive 1541 stampante 801** 500 supporti con più di 1500 programmi. Tolone Massimo via G. Galilei 16 - 35042 Monselice (PD) tel. 0429/ 74033

Vendo ZX Spectrum 48K + tutti gli accessori + il manuale d'istruzione + circa 70 programmi di giochi e utilities + il libro "77 programmi per lo spectrum" + tastiera professionale DK'Troniks con tastierino numerico e spazio per l alimentatore (usata circa 3 ore) il tutto a lire 450 000 trattabili se in contanti. Telefonare a Giampaolo Montini 45100 Rovigo · Tel. (0425) 31959 ore 16-21

Vendo **TI 99/4A + sintetizzatore vocale + minimemory + ext. Basic** + interfaccia stampante + personal record Kepping + personal report generator + household budget magement + home financial decision, tutti su moduli SSS. Vendo anche separatamente telefonare Daniele - Venezia 041/ 35305

Vendo **Commodore 64 + monitor 14 colori** (Commodore 1701) + **registratore + stampante Seikosha GP-100-VC** + 100 programmi vari e 75 riviste per computer e libri con centinaia di altri programmi + 2 Joy Stick. Tutto nuovissimo ancora imballato a lire 1 250 000 non trattabili. Telefonare ore pasti e chiedere di Marco al numero 041/ 26416 di Venezia

Vendo **APPLE II C + drive esterno + mouse** + dischetti originali, il tutto perfettamente funzionante, acquistato a settembre 84, ancora in garanzia fino al 9/85 pagato Lit 3 700 000 circa, vendo al modestis-

ATTENZIONE

Per gli annunci a carattere commerciale - speculativo è stata istituita la rubrica MCmicrotrade. Non inviateli a MCmicromarket, sarebbero cestinati. Le istruzioni e il modulo sono a pag. 193. Per motivi pratici, si prega di non lasciare comunicazioni o chiedere informazioni (telefoniche o scritte) riguardanti gli annunci inviati.

simo prezzo di lit 2 500 000 trattabilissimi Vendo anche separatamente purché alla svelta causa inaspettata partenza per il militare
Telefonare a Fabrizio, (010) 232072/ 208640

Vendo **TI 99/4A + Alim.** + Mod PAL + cavo per registr + Mod Video Chess (scacchi) + 3 casset. con circa 50 progr di vario genere + libro "Impariamo a programmare col TI 99/4A" Tutto con man d'uso e imballo originali Condizioni perfette 4 mesi di vita garanzia in bianco il tutto al favoloso prezzo di L 280 000 Solo zona GENOVA Telefonare ore pasti Pastorino Claudio via Roma 34/3 Masone (Genova) tel 010/ 926352

Eccezionale! Vendo Interface Joystick Programmabile per ZX spectrum, facilissimo da usare corredato da apposite istruzioni L 50 000 Inoltre vendo **tastiera DK Tronics con pad numerico per ZX Spectrum** L 150 000 il tutto usato pochissimo Telefonare ore pasti al 0182/ 90832 (spedizione in tutta Italia)

Vendo il più presto possibile per bisogno urgente di liquidi **Spectrum 48K** con tastiera professionale e più di 200 programmi (tantissimi giochi quasi tutti i linguaggi molti applicativi) a L 340 000, compreso manuale e diversi libri Telefonare (13 30/14 30) a Andrea Pacelli Via Firenze, 205 - 65100 Pescara Tel 085/26380

Vendo favolosa stampante per Spectrum **Seikosha GP 50 S** a lire 350 000 trattabili causa passaggio a sistema maggiore Due mesi di vita (come nuova) A chiunque dovesse comprarla regalerò 15 programmi tra i migliori per Spectrum 16/48 Regalerò anche tre rotoli da 60 metri e parecchi fogli di carta colorata Telefonare o scrivere a Pierluigi Iorio Via della Libertà Agropoli (SA) Tel 0974/822354

Vendo computer a colori **TI-99/4A** (completo) perfettamente funzionante + cavetto registratore + 2 joystick Texas Instruments + alcuni programmi a L 180 000 trattabili Vendo inoltre registratore Sanyo M1010 con alimentatore dotato di prese per cuffia, microfono, controllo a distanza adatto per TI 99 e Spectrum L 100 000 Francesco Quattrocchi Via Cesareo 24 - 98100 Messina Tel (090) 714514 ore pasti

Vendo per fine leasing **HP 86 + espansione 128K + monitor HP 82913A + doppio minifloppy HP 9121 + stampante grafica** 80/160 cps HP 82906A (in garanzia) con Visicalc File/80 Word 80 a L 6 000 000 Telefonare 080/216039 (Bari) pomeriggio

Vendo traduttore simultaneo **Sharp lq 3100** + modulo inglese, modulo tedesco modulo italiano + istruzioni in italiano e conversione misure Vendo tutto a L 200 000 Telefonare ore pasti allo 081/981214

Vendo traduzione dattiloscritta manuale programma Maserfile L 10 000 contrassegno **Enciclopedia di elettronica e informatica Jackson** L 180 000 (val comm 350 000) Natale Giovanni - Viale Trieste, 36 93100 Caltanissetta Tel 0934/22775 ore pasti.

Vendo **TI-99/4A** completo di alimentatore e modulatore TV + manuale + cavo registratore + registratore + modulo SSS scacchi + libri "Home computer TI 99/4A", "Imparate il Basic con il TI 99/4A", "Facile come il Basic" + 2 programmi per sistemi TO TO + giochi tutto per lire 200 000 (duecentomila) Scrivere o telefonare a. Piccirelli Nicola - Via Boccaccio, 4 66054 Vasto (CH) Tel 0973/51342

Vendo **Sharp MZ 721 completo interfaccia Centronics**, cambio software personalmene sviluppato p e simulatore di volo 20KB Infascelli Filippo - Via Galvani, 7 - Modugno (BA) Tel 564158

Vendo **VIC 20** nuovo, e **ZX 81 con espansione 16K** con i relativi manuali in italiano con 3 pubblicazioni della Jackson e inoltre n 3 cassette giochi, il tutto a sole lire 450 000 La Rocca Gennaro - Via Angilla Vecchia, 21 - 85100 Potenza

Vendo **per ZX Spectrum** gestione spese familiari in

modo dettagliato Può gestire fino a 400 spese al mese, completo di istruzioni e compatibile microdrive L. 30 000 Inoltre non avete tempo per i vostri programmi fatemelo sapere Per informazioni Francesco Del Gaudio - Via Salvatore Quasimodo - 87036 Rende (CS) Tel 0984/862743

Vendo **Spectrum 48K** tastiera professionale + Pad numerico + interruttore, con 100 ottimi programmi cavetti, manuali registratore tutto in ottimo stato In blocco L. 550 000 Franco Chiavetta - Via F Di Giorgi, 4 - 90145 Palermo. Tel 091/554264

Vendesi causa errato acquisto **Wildcard**, con SW e manuale **per Apple II, II+** o compatibili sblocca tutti i programmi protetti L. 80 000 Scrivere o telefonare a La Spina Salvo, Via Liberta 90 95019 Zafferana Etnea (CT) Tel 095/952174

Causa scarso utilizzo vendo nuovissima consolle **Atari VCS 2600 + 2 joystick** + cartuccia "Space Invaders" a lire 170 000 trattabili Telefonare ore pasti allo 0832/29783

Commodore 64 + unità a casette causa passaggio sistema superiore tel 080/941836 (chiedere di Stefano)

Affarone!!! Vendo **Atari VCS 2600** completo di cavetti, alimentatore due joystick due cartucce (Space Invaders e Battlezone) complete di dettagliate istruzioni d'uso Il tutto a sole L. 300 000 trattabili Lebano Michele - Via S Barbara, 46 Copersito Cilento (SA)

Causa passaggio a sistema superiore vendo **TI-99/4A** a L. 200 000, SSS (Extended Basic, Munch Man Alpiner Zero Zap ed altri) a L. 30 000 l'uno anche separtamente Telefonare ore pasti 0862/23995 chiedere di Francesco

Occasionissima **HP 87A** (MCmicrocomputer n 11) da 32K estendibili a 544 lire 2 800 000 doppio floppy da 540K HP 82901 M lire 2 600 000 Perfettamente funzionanti Come nuovi Anche singolarmente Ing V Devita Via M Serao 19 - S Giorgio a Cremano 80046 (NA) Tel 081 47957 ore pasti

Sono pazzo Vendo **TI-99/4A** + alimentatore + antenna + 2 moduli SSS Parsec scacchi + 2 cavi registratore + 5 cassette + libro delle istruzioni + 2 libri + joystick tutto L. 300 000 Ennio Fragano Via Tasso 254 b - 80127 Napoli Tel 081/668218

Causa passaggio a sistema superiore vendo **Apple IIe, n. 2 drive** di cui uno ancora in garanzia, monitor III con supporto Vendo inoltre software di oltre 50 programmi tra cui Apple Write IIe, Apple Post, Lock Smith utility, games gestionali, oltre a moltissimi programmi di Ingegneria Civile con manuali per un valore comm di oltre 6 000 000 Il tutto usato pochissimo a L. 4 500 000 Arch Marulo Luigi Tel 081/8621256 ore 21-22

Causa trasferimento vendo **sistema grafico Cascade 2** completo di plotter dischi rigidi tavolette grafiche Apple IIe e Disk 2 Drive, monitor Hantarex a colori ed altro completo del software per la gestione del progetto architettonico in 2 e 3D Il tutto in imballi originali Telefonare giorni dispari dalle 18 alle 21 a 081/8631338

Per programmare in Assembler il vostro TI-99, per sfruttare le grandi possibilità avete bisogno del **manuale Editor/Assembler** 500 pagine sull'uso del L. M nonché degli sprite, hi-res etc Scambio inoltre programmi in L. M per minimemory Scrivere a Noviello Roberto - Via M Cammarano, 10 80129 Napoli

Vendo **SHARP PC 1211 + interfaccia stampante** L. 300 000 (con manuali in italiano) Tel 051 276873 Mauro - ore pasti - preferibilmente vicinanze Bologna

Cambio

Scambio corpo macchina Canon AT.1 + custodia originale + obiettivo Grandangolare 28 mm. Tamron + filtro rosso + filtro Cross Screen + filtro polarizzatore + duplicatore di focale Tamron tutto in perfette condizioni **con ZX Spectrum 48 K** in buone condizioni. Scrivere a Gaetano Nieddu Via Bernini 29 - 07020 Fraz. Moneta - La Maddalena (SS)

Cambio programmi per CBM 64 Ne ho circa 550 di ogni genere (quasi tutti in L.M.) sia su disco che su nastro. Sono disposto a scambiarli anche con **programmi in L.M. per lo Spectrum**. Scrivere allegando lista o telefonare a Mascali Giuseppe Via R. Margherita 573 - 98028 S. Teresa Riva (ME) Tel 0942/791692

Per i commodoristi compro, vendo, scambio qualsiasi tipo di **programmi per 64 e VIC 20, Commodore 16** su disco o su cassetta. Scambio principalmente programmi per il CMD 64 con quelli del CMD 16. Telefonare ore serali a Marzano Renato 0964/66419 scrivere al seguente indirizzo Via Termicola II Trav nº 2 - 89034 Bovalino (R.C.)

Cambio **programmi per Commodore 64** (giochi utility/varie) solo su cassetta. Per informazioni scrivere a: Curto Giuseppe - Via F.lli Plutino 31 88100 Catanzaro

Cambio/Vendo circa 100 **programmi per Commodore CBM 64**. Per informazioni richiedere lista a Dente Malagrino - Viale Einaudi 59 2/D - 70125 Bari. Telefonare ore pasti al seguente numero 080/335499

Cedo **Commodore 64 + Unità a cassette** in cambio di reflex e/o ottiche pentax. Telefonare dalle 18.00 in poi al 080/941836 (chiedere di Stefano)

Scambio **Software ZX Spectrum** (oltre 700 titoli), preferibilmente novità inglesi. Risposta assicurata. Massima serietà. Scrivere a Scimncalepore Giuseppe Via P. Emilio 50 70059 Trani (BA) o telefonare ore pasti al 0883/45682

Scambio software per CBM 64 su dischetti games e gestionali. Dispongo di oltre 1000 titoli. Scrivere inviando le proprie liste a De Renzio Modesto - Via Matteotti 132 - 70032 Bitonto (BA)

Amstrad CPC 464 cerco utenti per scambio programmi ed esperienze. Dispongo molti giochi anche listabili. Originali inglese. Data Base - Text Editor ecc. Allegare Lista Software - Rispondo a tutti. Ennio De Carlo - Via Avellino 6 - 73100 Lecce Tel 0832/51517

CBM 64 user cede i suoi 350 programmi + un eventuale conguaglio (max L. 50.000) in cambio di stampante usata MPS 801/2 purché perfettamente funzionante. Scrivere a Paolo Pallara P.za Salerno 3 73100 Lecce o telefonare allo 0832/54514. N.B. Annuncio valido solo per le province di LE-BR-TA.

Cerco per Commodore 64 giochi ed utilities di ogni tipo, sia in Basic che in L.M. cambio eventualmente con oltre 200 giochi tutti in L.M. Cambio purtroppo solo su cassetta. Rispondetemi inviando la vostra lista, troveremo un accordo. Albanese Roberto - Via Francesco De Mura 40 - 73100 Lecce - Tel 0832/591221

Cambio programmi per Commodore 16 e Plus/4 su disco o su nastro. Cerco solo programmi di qualità. Richiedere lista a Marra Gerardo - Via P. Baratta 91/A - 84091 Battipaglia (SA) - Tel 0828/23130

Apple II. Scambio di software e notizie Hardware Posseggo numerosi programmi di Utility linguaggi e applicativi. Scrivete per avviare lo scambio a Castellone Gaetano Via Gerolomini 89 60078 Pozzuoli (NA)

Buonasera! Non vendo non compro. Ho 16 anni e ho scoperto che i **programmi per il C64** costano troppo per le mie possibilità economiche. Così ho pensato di affidarmi a MC per trovare qualsiasi persona che sia disposta a scambiare con me. Ho circa 500 programmi, scambio anche manuale in italiano. Con grande pazienza attendo la vostra telefonata!! Palumbo Luigi Tel 081/660068 - V.le Gramsci 26 Napoli

Cambio il mio **Spectrum 48 K** con più di 200 programmi di vario tipo (games, utility...) con un MPF II o un Apple compatibile perfettamente funzionante. Mauro Zaccaro - Via De Gennaro 45 - 80125 Napoli - Tel 081/610827

Sono interessato allo scambio di **programmi** americani ed inglesi per il computer **Commodore 64**, massima serietà, scrivere a Dati Luigi - II Viale Melina 15 Portici (NA) 80055 oppure telefonare al 081/471703

Cambio/vendo **programmi per Apple II/IIe/IIc**. Dispongo di una vastissima biblioteca software. Scrivere o telefonare a Mario Cacciapuoti - Via M. Semmola 114 80131 Napoli - Tel 081/468375 - Rispondo a tutti con la mia lista.

Cambio oltre 90 **programmi per Spectrum 16/48 K** tutti nuovi e divertentissimi! Sono disposto anche alla vendita a prezzi modici. Telefonare ore pasti al 081/253715 oppure scrivere a: Caputo Luciano Via G. Miranda 32 - 80131 Napoli

Sono un possessore del computer **Commodore 64** e vorrei formare un meraviglioso club dove si possano scambiare programmi su disco, su cassetta, e su lista. Finalmente tutto questo a Roma. Telefonatemi ore pasti al numero 06/7940637 Pecis Emiliano Via Appia Nuova 572/F (Colli Albani) - Roma

Cambio consolle Intellivision e comprese 5 cassette + istruzioni d'uso + imballo cavetti di collegamento con qualsiasi pocket computer, oppure vendo solo l'intellivision ad un prezzo eccezionalissimo. Telefonare allo (06) 5137131 chiedendo di Enzo

Oltre 4.000 **programmi per CMB 64** posseduti cerco programmi per ZX Spectrum e MSX. Guglielmo Bacchetta C.P. 374 - 60035 Jesi (AN) - Tel 0731/56705 dopo le ore 20

Compro + Scambio qualsiasi tipo di programma per ZX Spectrum. Inviate le vostre liste avrete risposta entro massimo una settimana. Rispondo a tutti. Garantisco massima serietà e competenza. Capelli Corrado - Via Monti Sibillini 4 - 60035 Jesi (AN)

Effettuo ottime traduzioni dall'inglese di testi e manuali e istruzioni inerenti i computer Sinclair in cambio di programmi per Spectrum 48 K (o comunque a prezzi modicissimi). Per informazioni pregasi allegare francobollo. Rossano Mariotti - Via E. Curiel 7 61032 Fano (PS)

Cerco Utenti Casio PB 300 per scambio programmi e notizie su nastro o listati. Giorgio Rizzo S.S. 326 n. 19 - 53040 Gracciano di Montepulciano - Siena - Tel 0578/708247 sabato e domenica.

C64 Scambio di software, documentazione ed esperienze tra possessori di **Commodore 64**. Disponiamo di ricca biblioteca di programmi. Contatti diretti in zone Firenze, Grosseto, Terni. Rivolgersi a: Edoardo Lampis - Via Cadorna 21 - 50129 Firenze. Tel 055/494648

Cambio **500 programmi per Commodore 64** con materiale per windsurf. Scrivere o telefonare a Enzo Catalano, Via Cacialgi, 29 56025 Pontedera (PI) Tel 0587/54756 per lista e offerte

Cambio **programmi per CBM 64**, prego mandare lista programmi. Cambio inoltre, per materiale CBM 64 **joystick e cassette SSS del TI-99/4A**. Vendo TI-99/4A a L. 190.000. Nuovo. Scrivere a Battistu Ivano - Via Martiri della Libertà 14 - 34079 Staranzano Tel 0481/710805

Cambio **per Commodore 64, programmi di ogni tipo** in particolar modo giochi, disponibili sia su nastro che su disco. Inviare lista a Giovanni Cereda - Via Marco D'Oggiono, 23 - 22053 Lecco (CO) oppure telefonare allo 0341/367387

Cambio **software di ogni tipo per Commodore Vic 20** con espansione da 8 e 16Kb. Inviatemi le vostre liste. Cambio sia listati che cassette. Massima serietà!!! Scrivere a Rodolfi Massimo - Via Limbolotto, 11 24058 Romano di Lombardia (BG) Tel 0363-910216

Cambio **RTX VHF AOR AR 280** a sintesi PLL 140 - 150 a passi 5KHz, potenza commutabile 1/5W, 3 memorie, shift +/- 600 KHz, batterie NiCd entrocoat., antenna in gomma, caricabatterie, cavo alim. esterna, nuovissimo con computer **Apple II** o compatibile. Eventuale conguaglio o vendita solo RTX. Berteselli Roberto - V. E. Ponti, 52 - 20143 Milano. Tel 02/8136569

Cambio **per Commodore 64** i migliori programmi (centinaia) preferibilmente su floppy disk, zona Milano. Telefonare ore serali a Lorenzo/Marco Tel 4075556

Cambio **RTX 144 MHz portatile "Belcom"** (da 141 a 151 MHz) con stampante per Commodore 64 (zona Torino). Tel. ore serali 011/9408029 o scrivere a Fabrizio Terranova, Str. Pino 17/6 Baldissero Torinese (TO)

Cambio **programmi per Spectrum 16/48K**. Per avere l'elenco dei programmi scrivere a Ballarini Bruno via Magenta, 5 - 10012 Bollengo (TO)

Posseggo un Dragon 32/64 e chiedo contatti con altri dragonisti per scambio di programmi, idee, consigli. Io ne posseggo alcuni e sono disposto a scambiarli o venderli. Scrivere o telefonare (ore pasti serali) a Marco Garipoli - C.so Europa 201/2 Tel 010/368725

Scambio software ingegneria strutturale per PC IBM Studio tecnico C&A Rimini V.le Dandolo 35 Tel 0541/52335 uff. - 25765 ab.

Scambio programmi per CBM 64 sia su disco che su cassetta. Possiedo ultimissime novità. Annuncio sempre valido. Scrivere o telefonare a Sciuca Pierpaolo Vicolo Castelfidardo, 30 - 35100 Padova. Tel 049/660378

Scambio software per MSX, possibilmente su disco. Tratto solo in zona Milano. Cellini Massimo tel 8322690 (ore pasti)

Scambio per TI-99/4A, interfaccia parallela esterna che non richiede il Box esterno e il modulo SSS Hangman, con l'interfaccia RS 232 Card originale Texas e i relativi manuali; effettuo lo scambio possibilmente di persona, inoltre cerco disperatamente il P-Code Card e l'Editor/Assembler. Monforte Ferrario Leandro Via A. Alessandri, 5 24100 Bergamo Tel 035/347078 (ore serali dopo le 20.00)

Scambio programmi per il CBM 64 possibilmente su disco. Per informazioni scrivere o telefonare a Porzio Ivan Via M. Buonarroti 28068 Romentino (Novara) Tel 0321/60445. Telefonare possibilmente ore serali.

Scambio software per Apple Mac Intosh di qualsiasi tipo. Dispongo già di un discreto numero di programmi. Scrivere o telefonare a Marco Verdesca Res. Archi Milano 2 - 20090 Segrate (MI) Tel 02/2138347

Scambio per **Commodore 64 e Vic 20** programmi solo in L.M. e cartridge su nastro. Scrivete e mandate i vostri listati a: Luca Mansutti - Via M. Grappa, 1 33100 Udine. Tel 0432/44768

Vendo scambio **Programmi per Apple II, IIe, IIc.** Posseggo giochi, utility, grafica. Scrivere o telefo-

micro MARKET

nare (meglio) a Longoni Luca - Via Alfieri, 7 - 20052 Monza (MI) Tel 039/833198

Vendo 3 cassette per Olivetti M 10 "Imparo a programmare", "Impariamo a usare il computer", "5 esercizi di programmazione" al prezzo di L 8 000 oppure cambio con equivalenti cassette Commodore 64 Manuel Buzzi Tel 581647 Via Coroncini, 5 Milano

Vendo e soprattutto **cambio software e utility per ZX Spectrum**. A chi mi invia la propria lista invio la mia Possiedo anche le ultime novità. Scrivere a Andrea Ozzano - Via B Fenoglio, 10 - 12100 Cuneo Assicuro una risposta a tutti

Cerco utenti dell'home computer TI-99/4A per scambio di programmi ed esperienze Inviare propria lista programmi a Cubani Pierfranco Via Costanza 15 20146 Milano oppure telefonare al 436428 ore 18/21.

Vendo e cambio **programmi per il TI-99/4A** (listati, cassette) Enrico Castagnetti - Piazza Municipio, 4 13060 Valdengo (VC)

Cambio programmi per Apple II, Mac, IBM PC Fornisco la mia lista a tutti coloro che mi avranno spedito la loro Nel caso di novità interessanti posso anche acquistare Grazia Meconi - V. Gramsci 56030 Perignano (PI)

Compro scambio vendo solo in Toscana **programmi originali per Commodore 64** Massima serietà e rapidità Manicagli Alessandro - Via Di Pratale, 1 56100 Pisa

Cambio e compro programmi per Sinclair QL. Cerco inoltre possessori dello stesso computer nella zona di Parma per scambio informazioni Tel 0521/798357 oppure scrivere a Marchiani Francesco - Via Quasimodo, 10 43050 Chiozzola (PR)

So che stavate cercando questo annuncio! A tutti i possessori del **Sega SC-3000** ho in mano il vero L/M linguaggio macchina del SEGA (12/32K) Cambio tutto ciò che ho anche per pochi programmi vostri che non ho Prego anche i clubs di contattarmi Rispondo a tutti Marco Sola - Via Secchia, 38 - 41056 Savignano S/P (MO) Tel. 059/760792

Cambio VIC 20 originale con espansione 8K con software contabilità generale e contabilità semplificata e relative istruzioni per C 64 Scrivere o telefonare a Lucia Roberto - Via S Luca 15/8 61124 Genova Tel 010/296436

Programmi per Commodore 64 esclusivamente su floppy ne possiedo oltre 300 Telefonare dalle 18 alle 19,30 Roberto Bucci Tel 8690425 Milano

Compro

Compro fotocopia **traduzione in italiano manuale del drive VIC-1541** e traduzione foglio istruzioni progr magic desk 1 commodore Accetto lista e prezzi programmi gestionali per acquisti Ferraris Carletto Via Noca 29 Valenza (AL) Telef (0131) 955213

Compro **manuale** o istruzioni in italiano del **programma stat 64 per commodore 64** D Gattino Via Susa 105 10096 Leumann TO

Compro **Sinclair ZX-81**, in buone condizioni e a prezzo modico. Compro anche qualche libro o programma per lo stesso Per informazioni scrivere o telefonare (ore pasti) a Carlo Campanini Via Campo dei Fiori 13 - 32021 Agordo (BL) 0437/ 65090

Compro **HP86-B** se vere occasione con accessori Rom e Ram Scrivere Dott Giorgio Rossetti Via Pelacani 2 43100 Parma.

Compro **Modulo EX. Basic** per il fantastico TI 99/4A Caleffi Cristiano Via Porrettana 78/7° Pontecchio Marconi "Bologna" 40044 Tel (051) 846396 ore pasti.

Compro **Modulo EXT. Basic per TI 99/4A** ad un prezzo massimo di L. 130 000 purchè funzionante Giustarini Paolo Via Norvegia 10 Pisa Tel 050/ 572468

Compro **per Apple II e programmi scientifici** di livello universitario Scrivere a Antonio Russo Poggio Giubbiani 100 50066 Reggello (FI)

Compro **espansione di memoria da 16K Ram per VIC-20** solo se in buono stato e a un prezzo accessibile Telefonare allo 071/ 9170375 e chiedere di Gianluca

Compro **registratore originale per commodore 64** solo se in perfetto stato e a prezzo ragionevole solo zona Roma. Alessandro Tel 5000251 - ore pomeridiane.

Compro **ZX Spectrum** non riparabile per recupero componenti Filippo 06/ 8444751 Telefonare dalle 18 alle 20

Compro **stampante compatibile con il TI 99/4A** tramite interfaccia Rs 232 Per accordi scrivere o telefonare a Conte Antonio Via Duca D Aosta n°67 80026 Casoria (NA) Tel 081/ 7580141 (ore serali)

Compro **per Apple II e programmi su disco** di giochi e utility Inviare la lista dei programmi con i relativi prezzi e modalità di pagamento a Montixi Sandro Via Pitzolo 26 - 09100 Cagliari

Compro o **cambio programmi per Spectravideo SV-328** ed in generale per standard MSX su listato o cassetta A richiesta ricambio con programmi per qualsiasi Computer Ricerco inoltre adesioni per un Club per Spectravideo Posseggo numerosi giochi e compilatori in Basic su cassetta per programmare in Forth e Assembly Scrivere prestissimo a Russillo Antonio - Via Lucio Petrone 77 - 64100 Salerno

Compro - scambio **software per CBM 64** solo novità 1985 in particolare videogiochi e avventure Inviare lista o richiedere la mia a Sergio Poli - Via Gustavo Modena 35 - 35100 Padova. Tel 049/ 850026

Cerco **Olivetti M24 e/o M21 d'occasione** oppure singoli componenti (video, tastiera, ecc) Scrivere o telef Salvo Cannici V le Trieste 108 93100 Caltanissetta Tel 0934/ 26491 32474

Cerco programma **"Flight Simulator II°" della Sublogic per CBM 64** Sono disposto a scambiarlo con alcuni dei miei programmi (oltre 200), su disco Scrivere a Lombardelli Roberto - Via Einaudi n°4 29010 Alseno (PC)

Cerco per **TI 99/4A Modulo "Speech Syntetizer"** Urgente Scrivere o telefonare a Sedda Roberto C so Casale N 10 Trino (VC) Tel (0161) 828240 (ore cena)

Cerco **programma Logo su disco per Apple IIe** con se possibile, manuale Telefonare dalle 19 00 alle 21.00 a. Luca Tel. 02/ 6080493 Milano

Compro **Kit trasformazione Floppy Disk CBM 3040 in 4040** scrivere Rinaldi Domenico - Via O Vigliani 15/ 10 10135 Torino

Cerco disperatamente **scheda CP/M per CBM 64**. Pago fino a lire 75 000 Cambio Prg per lo stesso ultime novità supercione, bozo s night, impossibile mission ecc Vendo prg. per l'elaborazione di sistemi totocalcio scritto da me (79 blocchi). Paolo Donalisio Via Torino 162 - 12038 Savigliano (CN) Tel 0172-33323

Acquisto **modulo "SSS Scacchi" per TI 99/4A** funzionante L. 30 000 Scrivere a Gioacchino Dimitri Via Marabitti 37 90142 Palermo

Cerco le seguenti **riviste MC Microcomputer** dal n 1 al n 30 e **"Personal Software"** dal n 1 al n 18 Compro non necessariamente in blocco, e purchè le riviste siano in buone condizioni ed a un prezzo ragionevole Pietro Garita - Via F sco Speciale n. 13 90129 Palermo - Tel. 091/ 481031.

Cerco e **compro per TI/99/4A modulo Extended Basic** con relativo manuale, moduli SSS Soccer, Ti Invaders, The Attack, Alpiner Othello Scrivere o telefonare (dopo le ore 21 00) a Laprocina Mario Viale Fortore 14 71100 Foggia Tel 0881/ 76841

Per **"Commodore 64" compro stampante in ottimo stato** Solo Puglia Rivolgersi a Bruno Cosimo, Viale della Libertà, 124 74015 Martina Franca (TA)

Acquisto **per TI 99/4A Modulo SSS EXT. Basic-Soccer** Inoltre coppia Joystick a prezzo interessante Scrivere a. Castigliola Vincenzo - Via Salute 33 80055 Portici (NA)

Cerco urgentemente, al prezzo di L 350/ 400 000 max , **Disk drive esterno con Controller per TI 99/4A**, Editor/Assembler + manuale L 50 000 sintetizzatore vocale L 50 000 max Terminal Emulator II° 20 000 max Scambio software basic ed extended in listato Spedire lista vostri programmi Toffali Tiziano - V Ruggero Leocavallo 3/F - 37131 Verona Tel 045/ 521642

Per TI 99/4A periferical box cerco buon prezzo Rs 232 ed espansione 32K - interessato anche miaime mory - tratto preferibilmente Veneto Telefonare ore serali 0444 - 530038 o scrivere a Alberto Guffrida Via P Lioy n° 8 - 36057 Arcugnano (VI)

Acquisto **modulo texas instrumens TI Extended Basic** max L. 100 000 Solo Milano e provincia Adriano Guarneri Via Cesare Battisti, 33 - 20061 Carugate (MI) Telefono ore ufficio 02/ 875646

micro MEETING

Annunci gratuiti per richiesta di contatti e scambio di opinioni ed esperienze tra privati.
Vedere istruzioni e modulo a pag. 193.
Per motivi pratici, si prega di non lasciare comunicazioni o chiedere informazioni (telefoniche o scritte) riguardanti gli annunci inviati.

Disponendo di un **personal computer IBM** contatterei persone residenti in Milano per scambio di idee e software Se interessati chiamare Riccardo 02/4233024

Possedete un SC-3000 e non sapete niente su questo Home e non avete programmi? Non scoraggiatevi Tenetelo C'è chi pensa a voi Aderite all SC Club chiedere non costa niente (almeno il francobollo per la risposta) Perico Romano - Via Geroni n 2 24025 Gazzaniga (BG) Tel 035/711993

Il TI/soft/59/club desidera entrare in contatto con altri club di utilizzatori TI/59 per un ampliamento dei rispettivi elenchi programmi o una eventuale fusione o associazione
Scrivere inviando informazioni elenco programmi e altro a TI/soft/59/club c/o Rimicci Andrea TS59C Via delle Fornaci 12/1 17011 Albissola Capo (SV)

Sei un possessore del PC della Triumph-Adler? Molti amici hanno già aderito alla costituzione del "Club" Se ti interessa farne parte per cambio esperienze, programmi ecc (bioritmi - istogrammi - test - tris agenda telefonica archivio personalizzato - roulette ecc), scrivi per informazioni, allegando francobollo a Riva Giovanni - Via al Monte S 9/1 Albissola Mare (SV) Tel 019/26817 ufficio 019/46229 casa

Cerco **possessori Commodore 64** per scambio di idee esperienze programmi eventuale formazione di club Scrivete o telefonate a Siccardi Roberto, Via Matteotti 178/2 18038 S Remo (IM) Tel 0184/75687 (ore 15.00 - 19.00)

Si è costituito a Udine il "64 Wizards Club" gruppo di scambio esperienze software sul **Commodore 64** Scrivere per informazioni a Maurizio Doni Via Passons 38 Udine

Posseggo un **Apple IIe** da poco tempo e vorrei contattare in tutta Italia possessori di questo computer per scambio idee esperienze, software Fulvio Giacomello - Via Montenero 13/c - Montereale Valcellina 33086 (PN)

Vicenza Cerco possessori di **ZX Spectrum** per scambio Software ed esperienze Telefonare ore 16.30/18 00 - 0444/546461

Cerco **possessori di Amstrad CPC 464** a Treviso per scambio, esperienze, idee, programmi ed eventualmente la fondazione di un club Telefonare allo 0422/542295 o scrivere ad Andrea Riva Viale 4 Novembre 37 - 31100 Treviso

Golden Apple Club - Tutto quello che avreste voluto sapere sull **Apple** ma non avete mai osato chiedere (**IIE, IIC, Mac, Compatibili**) Per informazioni Golden Apple Club C P 596 - 35100 Padova

Contatto **possessori computer Amstrad CPC 464** per scambio informazioni ed eventualmente programmi Carlotto Luca - Via Franzela n 61 35100 Padova Tel 049/608569

Si è costituito in Provincia di Padova il 1° **club 64** Montagnana soft per scambi di idee ed esperienze programmi originali Telefonare allo 0429/82469 (Carlo Ferrari) Club 64 Casella Postale n 6 35044 Montagnana (PD)

Contatto **possessori di Apple II e IIe** per scambio esperienze e programmi inviatemi la vostra lista e vi risponderò con la mia Gaboli Ivan - Via XXIII Marzo 228/A 28100 Novara Tel 0321/400451 (solo sabato e domenica)

Desidero contattare hobbisti possessori di **commodore 64** Zucchi Alessandro - Trafiume 28052 Cannobio NO Tel 0323/70835

Desidererei entrare in contatto con privati o club per scambi di idee programmi Sono un possessore di un **Apple II E** Possibilmente Torino-cintura Scrivere o telefonare (ore pasti) Audone Carlo - Via Marco Polo 41 - 10129 Torino Tel (011) 599587

Cerco **possessori di TI 99/4A** (preferibilmente zona Torino) per scambio di idee e software (posseggo l Extended Basic) Alberto Carazza Via Lodi 2 10152 Torino - Tel 237041

Apple User - Sono interessato ad entrare in contatto (**anche via modem**) con possessori di **computer Apple** Possiedo n 1000 programmi Moretti Roberto St Torino 73 10024 Moncalieri

Cerco **possessori di CBM 64** per scambio di programmi Inoltre vendo o acquisto programmi di Utility o giochi Questi ultimi solo se in l M Interessato anche a contattare eventuale "Commodore Club" nei dintorni di Milano Per informazioni e/o lista miei programmi scrivere a Stefanoni Roberto Via Torazza 152 - Brugherio (MI) oppure telefonare 039/879122 ore pasti

Desidero mettermi in contatto con eventuali possessori attuali e futuri del computer (Q L) della **Sinclair** per scambio informazioni e software su questo fantastico computer Scrivere a Tuscano Francesco Via Salici 17 20025 Legnano (MI) Tel 0331/597054

Cerco possessori presenti e futuri di **Quantum Leap Sinclair** per fondare un QL User-Club e scambiare idee e software - Ghezzi Roberto - Via Volontari del Sangue 202 20099 Sesto San Giovanni MI Tel 02/2485511 ore serali

Laureando in fisica cerca consigli circa l eventuale acquisto di scheda **Saybrook M68000 per Apple II +** chi ha notizie esperienze dirette o consigli può scrivere se vuole a Massimo Brambilla - V le Monterosa 106 - 20043 Arcore (MI)

Annunciamo primo **IBM PC club user group** Italiano il cui scopo è raggruppare gli utilizzatori dell IBM PC e compatibili per lo scambio reciproco di esperienze hardware e software Per proposte soci scrivere a IBM PC User Group Casella Postale 59 20019 Settimo Milanese (MI)

Scambio esperienze e programmi con possessori di **Apple IIe** con memoria centrale superiore a 128 K in particolar modo software che sfrutti l espansione di memoria, inoltre sono interessato ai programmi che utilizzano il Mouse II Francesco Pezzino Via Consolé Marcello 25 20156 Milano - Tel 02/365365

IHT Division Users. l'unico Club italiano per **Commodore 64** riconosciuto dalla Commodore americana e da autorevoli riviste Iscrivetevi subito per tutti i soci incredibili vantaggi, bollettino mensile accesso alla softbank utenza via Modem della nostra banca dati. Non perdete questa occasione Nessuno scopo di lucro IHT Via Borgonuovo 19 - 20121 Milano

Cerco programmi Binari utilities e notizie utili per un uso migliore del personal computer **HP 86 B** Roberto Giulietti Via F De Vico 16/E - 00143 Roma - Tel 06/5920440

Il Sinclair Club Roma Talenti per ampliare il proprio club cerca **possessori di Spectrum** in tutta Italia tra coloro che vogliono usare il computer come videogioco sofisticato o che cerchino qualcosa di più Per maggiori informazioni sulle attività del club o sulla iscrizione rivolgersi a D Ascenzo Massimo Via F D'Ovidio 109 00137 Roma Tel 06/8280043 (ore pasti)

Contatterei **possessori Ti-99/4A** per scambio di software Inviatemi la vostra lista ed io farò altrettanto Scrivete o telefonate a Gino Luerezi Via dei Giardini 18 67100 l Aquila - Tel 28307

Cerco contatti per scambio idee e programmi per la serie **Apple II** Carmine Perantono Via Perruna 58 66010 Tollo (CH)

L'Istituto "Opera per la salute del fanciullo" Via Bellaria 4 - Napoli è interessato a visionare e valutare eventuale **collaborazione per software di tipo didattico** per la fascia della Scuola Materna ed Elementare implementabile su commodore Vic 20 e CBM 64 Tel 081 7418802 ore 10-16

Possessore di **ZX Spectrum** cerca listati di programmi di ingegneria civile (strutturale) per Apple e Commodore da convertire Mauro Zaccaro Via De Gennaro 45 - 80125 Napoli Tel 081/610827

A tutti i possessori di ZX Spectrum 16/48 - Plus e CBM 64 Attenzione!!! Sta nascendo a Bari il "Computer Center" siete tutti invitati ad iscrivervi Per informazioni telefonare al 080 41-46-80 o scrivere a Ciannamea Fabrizio C.so A De Gasperi 294/A Scala D 70125 Bari

Desidero contattare possessori del computer **Decision Mate V** per scambio programmi ed esperienze telefonare a Nibile Giuseppe Via Manzoni 128 Agrigento Tel 0922/55185

Cerco **possessori di ZX Spectrum 16/48k** per scambio di idee software e hardware in particolare con radioamatori (solo OM) per scambio esperienze Invia la tua lettera lista a IT9XXS Giovanni Mazzola Viale Strasburgo 111 90146 Palermo Tel 091 501691

Cerco possessori del **Commodore 64** per scambio di idee e programmi Dispongo di circa 250 programmi in prevalenza giochi Sarei interessato a utilities Per ogni informazione telefonate dalle ore 18 in poi a questo numero 0789 22030 Oppure scrivete a Ruiu Giancarlo Via Giordano Bruno 17 07026 Olbia (SS)

Cerchiamo **Utenti ZX Spectrum** per scambio idee-programmi Non aspettate Telefonate al Microclub Ad ogni socio 3 titoli in omaggio Tel 070/659784 Via De Magistris 4 09100 Cagliari

È nato il Woodstock Sinclub Sassari, per tutti gli **utenti Spectrum** della zona Chiunque fosse interessato è pregato di telefonare al 079/216278 chiedendo di Andrea

Cerco possessori di computer Sinclair in Parma e zone limitrofe per fondazione club. Richiedo la massima Serietà. Bettati Andrea - Via Righi 3 - 43100 Parma Tel. 93365 ore pasti

Cerco possessori di Advance 86 A 16 bit, per scambio di informazioni sul sistema, sul Advance 86 Basic, e di programmi applicativi di ingegneria strutturale. Ricerco inoltre il modo di utilizzare da Basic tutta la memoria Ram oltre i 65 K Bytes. Scrivere o telefonare a ing. Albertazzi Andrea - Via B. Gigli 16 - 40137 Bologna - 051/440892.

Cerco possessori di **programmi per il QL** sia giochi che linguaggi ed utility. Scrivere o telefonare a: Baldassarri Gianluca - Via Marcolini n. 9 - 47100 Forlì Tel. 0543/32544

Desidero contattare possessore del **Sega SC-3000** per scambio di idee, esperienze e programmi. Scrivete a Tassinari Gianni - Via Sant'Andrea 4 - 47010 - Carpinello (FO)

C-64 Club. Organizzazione operante in Toscana ed in Umbria per lo scambio di documentazione, esperienze e programmi tra **Utenti Commodore 64**. Scrivere a C64 Club - Edoardo Lampis - Via Cadorna 21 50129 Firenze

Potendo utilizzare un **Macintosh**, scambierei opinioni ed esperienze con persone interessate. Berretta Maurizio - V. Aeroporto 63/C - 56100 Pisa

Cerco possessori del personal **Casio FP-1000/1100** per scambio esperienze ed eventuale software. Scrivere a Cesari Marco - Via D. Alighieri 29 - 52037 Sansepolcro - Arezzo - o telefonare allo 0575/734505

Desidererei contattare **possessori di Dragon 32** per scambio di idee, esperienze e programmi. Scrivere a Capitani Claudio - Via A. Costa n. 8 - 58100 Grosseto

Cerco possessori **Commodore 64** Ancona e zone limitrofe per scambio esperienze e programmi. Scrivere a Lanari Raniero - Via A. Costa 8 - 60128 Ancona o telefonare allo 071/871077

Possiedi un CBM64 o un C16? Non puoi assolutamente non far parte del nuovissimo club "Program Commodore" costituitosi ad Offagna (AN). Per iscriverti invia un tuo programma a: Gabriele Gardella - Via Viola n. 3 - 60020 Offagna - Tel. 071/7107119 (pregasi telefonare dalle ore 15 alle 18)

Apple Users Club - Macerata scambia/vende programmi per Apple II, IIe IIc anche sviluppati dal club per gli utenti locali. Tutti i programmi sono provati (e quindi funzionanti) ed accompagnati da manuale o istruzioni. Rivolgersi a Roberto Venturi - Via Ugo Foscolo n. 14 - 62100 Macerata

Se avete scoperto anche voi il pianeta **MSX** scrivetemi, vorrei contattare ragazzi e ragazze sui 16 anni magari **zona Umbria** per scambio programmi. Insieme faremo cose da pazzi. Andrea Dentini - Via Tiberina 78 - Todi (PG) - Tel. 075/882792

Desidero contattare possessori di **ZX Microdrive** per scambio software ed informazioni. Andrea Alberghina - C. Clodia 29 - 00195 Roma MC

micro TRADE

Annunci a pagamento di carattere commerciale-speculativo fra privati e/o ditte: vendita e realizzazione di materiali hardware e software, offerte varie di collaborazione e consulenze, eccetera. Allegare L. 30.000 (in assegno) per ogni annuncio. Vedere istruzioni e modulo a pag. 193.
Non si accettano prenotazioni per più numeri, né per più di un annuncio sullo stesso numero. MCmicrocomputer si riserva il diritto di respingere, a suo insindacabile giudizio e senza spiegazioni, qualsiasi annuncio dietro semplice restituzione della somma inviata.
In particolare saranno respinte le offerte di vendita di copie palesemente contraffatte di software di produzione commerciale.
Per motivi pratici, si prega di non lasciare comunicazioni o chiedere informazioni (telefoniche o scritte) riguardanti gli annunci inviati.

Per IBM PC e Olivetti M 24 vendo programma di totocalcio da me realizzato, con possibilità di riduzione colonnare in base ai segni e agli errori, semplice da usare e molto veloce. Il programma permette la registrazione del sistema, la ricerca della colonna vincente, la stampa su tabulato o su schedina, archivio squadre con dati e punteggi. Costo L. 300.000. Su richiesta si invia dimostrativo. Scrivere o telefonare a: Roberto Livi - Via Bissolati, 5 - 61100 Pesaro - Tel. 0721/65873

Vendiamo ottimi **programmi** (gestionali, utility, educazionale, grafica, ecc.) **per Commodore 64** completi di manuali. Disponiamo per **Hewlett Packard 150** di: Visicalc, Multiplan, Microplan, Lotus, Simphony, DBase II, MBA Context, Basic, Cobol, Fortran, Pascal, Wordstar, Condor, Delta, Pert, statistica, contabilità, magazzino, condominio, studi medici. Richiedere Soft Guide 64/150 a: Pocket Group - Via M. Amoruso, 34 - 70124 Bari

Commodore 64, Sharp 700, tutti gli MSX, Commodore 16 e 4 plus, Amstrad, CBM 8000, assortimento enorme di **programmi** originali e di produzione propria. Per 64 ultimissime novità di giochi, gestionali e utility. Per Sharp 700, MSX, CBM 16 e Amstrad eccezionale serie di programmi di produzione propria e tantissimi giochi a prezzi eccezionali. Computer House di Giovannelli Claudio - Via Ripamonti, 194 - 20141 Milano - Tel. Ab. 02/536926 - Tel. uff. 02/563105

Tastiere ZX Sinclair, massima velocità di battuta, scegli la soluzione più congeniale, da L. 49.000. **Quadrusopp** il quadruplo alimentatore che protegge anche il computer, L. 23.000. **Omnicard 55** orologio in tempo reale, Joystick, porta I/O e altre virtù, a sole L. 68.500. Inviando il bollo, riceverete ampia documentazione. Microcyber Piazza Isei, 28 - 47023 Cesena - Tel. 0547-20890

Vendo, cambio **software per Apple II, IIe, IIc**. Programmi di ingegneria, contabilità, data base, utilities, games, ecc., provvisti dei relativi manuali - vasta scelta - ultimissime novità. **Personalizzazioni di programmi a richiesta**. Gianluca Pomponi - Via Raffaello, 5 - 56020 Casteldelbosco (PI) - Tel. 0571/488012

IBM PC User Group Italia associato al più grande IBM PC User Group del mondo (New York) accetta iscrizioni utenti PC e compatibili al fine di rivelare i segreti più nascosti del vostro 16 bit. Scrivete a: IBM PC User Group Italia - P.O. Box 59 - 20019 Settimo Milanese (MI)

Il "Gruppo Utilizzatori Computer - Napoli" annuncia l'apertura della nuova sezione dedicata agli MSX che si affianca alla già lanciatissima sezione Sinclair. Se possiedi uno **Spectrum** oppure un **computer MSX** scrivici. Potrai far parte del club ed entrare in contatto con centinaia di amici in tutta Italia! Scrivici allegando due bolli lettera, avrai anche l'adesivo in regalo! G.U.C. Napoli, c/o Dott. Roberto Chimenti - Via Luigi Rizzo, 18 - 80124 Napoli Tel. 081/617368 (prenderà il 7623121)

Per Apple IIe 80 colonne vendo **programma di contabilità per studi medici** di piccola entità. Per Macintosh vendo un potente copiatore, inedito, e molti altri non in commercio. Telefonare o scrivere a: Luigi Palumbo - Via A. Ristori, 8 - 00197 Roma - Tel. (06) 802783

Vendo **totocalcio ad altissima velocità** (L.M.)

per ZX-Spectrum 48K 254 livelli di condizionamento, consecutività, verifica risultati, ridotti e biridotti, raggruppamento colonnare, manuale istruzioni Per informazioni Mario Pacchiarotti - Via Tuscolana, 884 - 00174 Roma Tel. (06) 7612788 (ore serali)

Per Apple o compatibili vendo **programma per gestire i risultati delle elezioni** politiche, comunali, provinciali, regionali In tempo reale esegue percentuali e raffronti con elezioni precedenti Per le elezioni comunali esegue immediatamente la ripartizione dei seggi anche se non tutti i dati sono disponibili In tal modo sin dall'inizio dell'arrivo dei dati, si possono eseguire delle previsioni e proiezioni Scrivere al prof Massimo Spezialetti - Via Probi, 2 64032 Atri, oppure telefonare allo 085/87362 ore pasti

Studio Ing Cosimi - Via Lucania, 1 - Siena Tel (0577)48131, unica organizzazione per la **distribuzione di software tecnico e vario per CBM 64** a professionisti, ingegneri, architetti, geometri, periti edili, comprensivo di tutta l'assistenza necessaria e la consulenza telefonica per i casi più difficili Programmi anche a richiesta Solo su Commodore 64 e 8096

DATA 99: una stupenda raccolta di 18 **programmi inediti** per il tuo **Texas TI-99/4A** interamente scritti in TI-Basic In questo numero troverai programmi di svago (Safari, Duello, etc), di utilità (agenda auto, arch cassette ecc), didattica (Italia — con una perfetta riproduzione della carta geografica) Inoltre uno stupendo Adventure ispirato a "La Storia Infinita" La cassetta più il manuale ti verranno spediti in contrassegno a sole L. 20.000 (comp spedizione) Si assicura tempestività e massima serietà Per richiedere il primo numero DATA 99 - Via Arenella, 81 - 80128 Napoli Tel 081/240796

Tutti vendono computer, noi vendiamo il migliore **software per Spectrum e Commodore 64** Personalizziamo i programmi a richiesta Arrivi settimanali da tutto il mondo Copiatori sofisticati per ogni necessità Giochi meravigliosi Ingegneria tutti i titoli con manuali Chiedere lista dettagliata a Bifolchi Giordano Via G nel Corso, III - 53045 Montepulciano (SI) Tel 0578-757907/716397

Vendo cambio oltre 2000 programmi gestionali (contabilità, magazzino ...) di ingegneria (strutturale ...), di gioco (novità), animazione, totocalcio, utilità (linguaggi ...) per i calcolatori Commodore 64 - Plus 4 - C16 - apple II, IIe, IIc - HP 86 - 87 - IBM PC - Spectrum QL Vendo interfacce stampante e modem (novità) per CBM 64 - Joystick programmabile + Kempston per Spectrum - Ing Maurizio Carola - Via L Lilio, 109 - 00143 Roma - Tel 06/5917363

Sviluppo e vendo **programmi originali di ogni tipo per Apple II, IIe, IIc**, oltre 1000 titoli disponibili con manuali Richiedere elenco a Oggioni - Via Gonzales, 4 - 20139 Milano Tel. 02/5397867

Novità!!! ora con il tuo computer puoi anche sviluppare le espressioni algebriche Il programma viene fornito completo di manuale d'uso ai seguenti prezzi per CBM 64 L 33 000 Per VIC 20 con almeno 20K L 26 000, per Spectrum 48K lire 27 000 La spedizione è effettuata contrassegno Prenotatevi Montalto Aldo - Via Zupi, 19 - 87100 Cosenza

Amstrad Computer Club Italia mettetevi subito in contatto per collaborare sapere conoscere Nessuna tassa di iscrizione Già disponibili nel club tutti i manuali, gli schemi e oltre 300 programmi di ogni genere Già in distribuzione il notiziario "Tutto Amstrad" 50 pagine di listati, informazioni, Scrivi Francomputer Club - Corso Fogazzaro, 139 - 36100 Vicenza

TI-99/4A - file a indice Potente e velocissima procedura di gestione **Archivio con chiavi** (1000/2000 rec) Richiede Ext basic, Mem. Expansion, Disk Drive, stampante (opzionale) Dettagliato manuale d'uso Ideale per magazzini, anagrafici librerie ecc Realizzazione professionale e funzionamento in tempo reale Molteplici funzioni per la più completa e sicura manipolazione dei records **Riorganizzazione automatica dell'archivio indice** Prezzo L 30 000 + sp post Scrivere o telefonare a Strafile Alberto - Via Zuccoli 19 - 20125 Milano Tel 02/6896739

Per Apple II/IIe vasto assortimento di accessori ed interfacce, garanzia 6 mesi Computers, joystick, drives, monitors, sintetizzatore vocale, penna ottica, schede 80 colonne, IC test, interfacce parallele e seriali, Z80 CP/M, programmatore di eprom, stampanti, modem Tramer - C so S Martino 0/H - 10122 Torino Tel 011/519505

Per Commodore 64 vendo ottimi programmi su disco o cassetta, cartridge con utility, molti manuali, pulsanti reset, circuito per attivare 2 registratori senza il 64 Richiedere lista gratuita Cassette per collezionisti di musica, totocalcio, giochi d'azzardo Fanelli Gabriele - Via C Zaccagnini, 129 - 00128 Roma Tel 06/6151345

Per Commodore 64, VIC 20, C 16 vendo scheda per duplicare in maniera digitale qualsiasi programma L. 40 000 **Vendo programmi** di giochi (novità), gestionali, ingegneria, utilità (grafica, copiatori) per Commodore 64 (oltre 1800) Apple, IBM PC, Commodore serie 8000, Plus 4, C 16, QL, Spectrum Vendo modem ad accoppiamento diretto per CBM 64 (novità) Interfacce Joystick programmabili Kempston, con 2 joystick L Carotenuto - Via Antonio Ciamarra, 26/e - 00169 Roma tel 06/7402032

Vendiamo per Commodore 64 oltre 1500 programmi di tipo gestionali contabilità generale, semplificata, magazzino, fatturazione, condominio, modello 740, etc , ingegneristici, matematico/statistici, totocalcio, grafica, copiatori, giochi **Catalogo a richiesta** con 200 programmi altamente selezionati Novità dagli USA e dall'Inghilterra a prezzi concorrenziali Scrivere o telefonare a Salvatore Salamo C.so Mazzini, 28 (CS) Tel (0984) 74150 Massimo De Luca - Via Neghelli, 13 (CS) Tel

MicroMARKET
MicroMEETING
MicroTRADE

Desidero che il presente annuncio venga pubblicato nella rubrica

☐ **Micromarket** ☐ vendo ☐ compro ☐ cambio — Annunci gratuiti per vendita o scambio di materiale usato o comunque in unico esemplare fra privati

☐ **Micromeeting**
Annunci gratuiti per richiesta di contatti e scambio di opinioni ed esperienze tra privati

☐ **Microtrade**
Annunci a pagamento di carattere commerciale-speculativo fra privati e/o ditte vendita e realizzazione di materiali hardware e software originale, offerte varie di collaborazione e consulenze, eccetera. Allegare L. 30.000 (in assegno) per ogni annuncio (lunghezza massima spazio sul retro di questo modulo). Non si accettano prenotazioni per più numeri né per più di un annuncio sullo stesso numero

Per motivi pratici, si prega di non lasciare comunicazioni o chiedere informazioni (telefoniche o scritte) riguardanti gli annunci inviati

MCmicrocomputer
RICHIESTA ARRETRATI
39

Cognome e Nome ....
Indirizzo ....
C.A.P. .... *Città* .... *Prov* ....

(firma) ....

Invietemi le seguenti copie di MCmicrocomputer al prezzo di L. 5.000* ciascuna:
*Prezzi per l'estero Europa e Paesi del bacino mediterraneo **L. 8.000** - Altri (sped. via aerea) **L. 10.000**

Totale copie __________ **Importo** __________

Scelgo la seguente forma di pagamento:
☐ **allego assegno di c/c intestato a Technimedia s.r.l.**
☐ **ho effettuato il versamento sul c/c postale n. 14414007 intestato a: Technimedia s.r.l. Via Valsolda, 135 - 00141 Roma**
☐ **ho inviato le somme a mezzo vaglia postale intestato a: Technimedia s.r.l. Via Valsolda, 135 - 00141 Roma**

N.B.: non si effettuano spedizioni contrassegno

MCmicrocomputer
CAMPAGNA ABBONAMENTI
39

Cognome e Nome ....
Indirizzo ....
C.A.P. .... *Città* .... *Prov* ....

(firma) ....

☐ **Nuovo abbonamento a 12 numeri**
Decorrenza dal N. ....

☐ **Rinnovo**
Abbonamento n. ....

☐ **L. 36.000 (Italia) senza dono** ☐ **39.500 *con dono***
☐ **L. 80.000 (ESTERO: Europa e Paesi del bacino mediterraneo) - senza dono**
☐ **L. 116.000 (ESTERO: Americhe, Giappone, Asia etc., sped. Via Aerea) - senza dono**

Scelgo la seguente forma di pagamento
☐ **allego assegno di c/c intestato a Technimedia s.r.l.**
☐ **ho effettuato il versamento sul c/c postale n. 14414007 intestato a: Technimedia s.r.l. Via Valsolda, 135 - 00141 Roma**
☐ **ho inviato le somme a mezzo vaglia postale intestato a: Technimedia s.r.l. Via Valsolda, 135 - 00141 Roma**

Testo dell'annuncio **MC 39**

Attenzione: *gli annunci inviati per le rubriche Micromarket e Micromeeting il cui contenuto sarà ritenuto commerciale-speculativo e gli annunci Microtrade mancanti dell'importo saranno cestinati senza che sia data alcuna specifica comunicazione agli autori*
Per gli annunci relativi a Microtrade MCmicrocomputer si riserva il diritto di respingere, a suo insindacabile giudizio e senza spiegazioni, qualsiasi annuncio dietro semplice restituzione della somma inviata. In particolare saranno respinte le offerte di vendita di copie palesemente contraffatte di software di produzione commerciale
Per motivi pratici, si prega di non telefonare comunicazioni o chiedere informazioni (telefoniche o scritte) riguardanti gli annunci inviati.
Scrivere a macchina. Per esigenze operative, gli annunci non chiaramente leggibili saranno cestinati.
Spedire a: Technimedia - MCmicrocomputer - Via Valsolda 135 - 00141 Roma

Completa la tua raccolta di MCmicrocomputer

Compila il retro di questo tagliando e spediscilo oggi stesso

Spedire in busta chiusa a:
TECHNIMEDIA
MCmicrocomputer
Ufficio diffusione
Via Valsolda, 135
00141 ROMA

Ti piace MCmicrocomputer? Allora ABBONATI

Compila il retro di questo tagliando e spediscilo subito

Spedire in busta chiusa a:
TECHNIMEDIA
MCmicrocomputer
Ufficio diffusione
Via Valsolda, 135
00141 ROMA